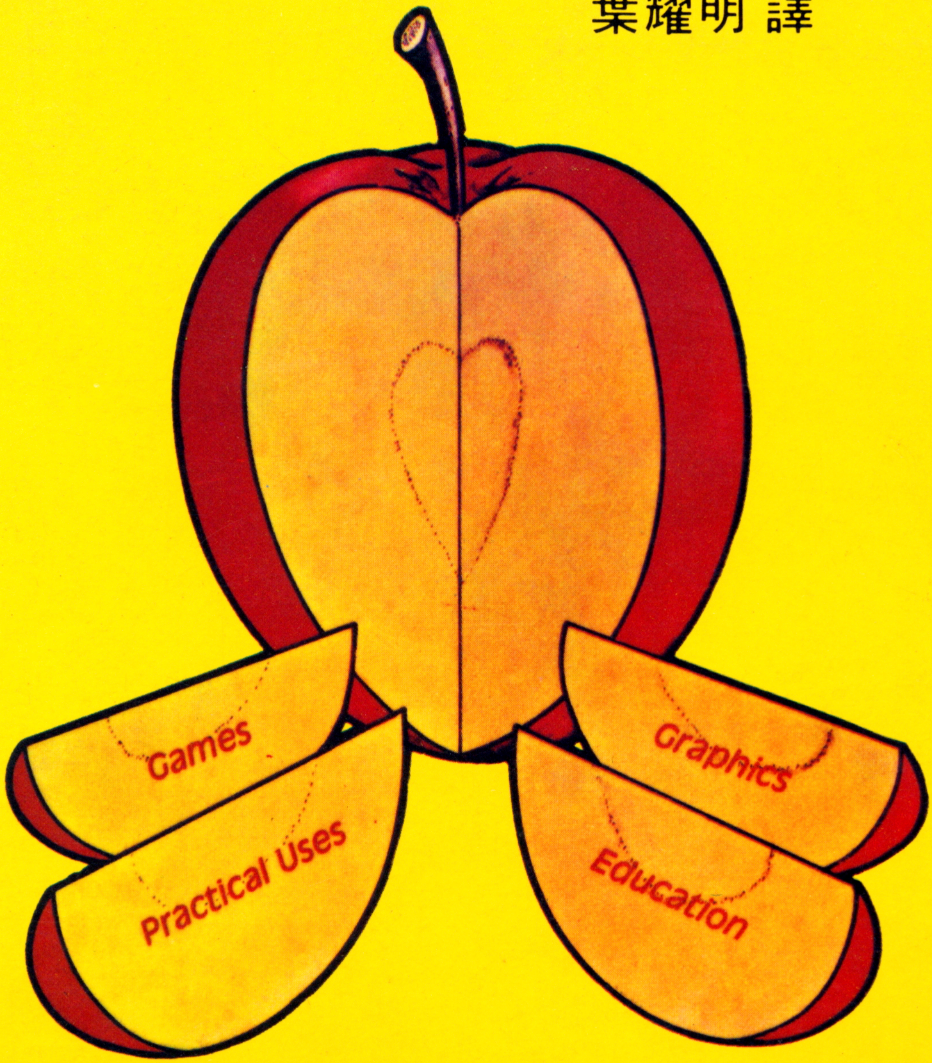
APPLE II
BASIC應用程式

葉耀明 譯



APPLE Ⅱ
BASIC應用程式

葉耀明 譯

協 群 科 技 出 版 社

APPLE II BASIC
應用程式

編譯者： 葉 耀 明
出 版： 協群科技出版社
發 行： 協群科技出版社
香港中環卑利街684號二樓
印刷者： 廣 源 印 務 局
青山道875號工廠大厦

定價： H.K.$ 35.00

如何使用本書

本書的每一章節提出一個電腦BASIC程式，其可在Apple Ⅱ或Apple Ⅲ 的Applesoft BASIC 上執行。大部分的程式可以在16K Apple Ⅱ上執行（見附錄1），每一章由八節所構成，每節的內容如下：

1. 目的：解釋程式的功能及你可應用的方面。
2. 使用方法：詳細說明你執行程式後所發生的現象，解釋你可做的選擇以及必須輸入的回答。指出程式的限制以及可能發生的錯誤。
3. 執行範例：顯示出執行程式後，可在螢幕上看到的結果。
4. 程式：印出BASIC程式本身。
5. 修改程式：告訴你如何對程式做一些簡單的修改，而得到不同的結果。你不須完全瞭解程式所有細節就能做此修改。
6. 主流程：解釋程式的流程邏輯。裡面包括BASIC程式的行號以及程式的工作。
7. 主要變數：說明程式內所使用的每一個變數用途。
8. 研究建議：提供跟程式有關的研究方向以及增進程式功能的技巧。欲做此項工作，你必須會

設計BASIC程式以及使用“主流程”和“主要變數”所提供的資料。

若只要使用你的Apple 電腦來執行這些程式，則只要閱讀前四節就夠了。後面四節是讓你瞭解程式內部以及擴充程式之用的。

建　議

以下爲提供各位如何配合本書的程式之幾項建議:

1. 先閱讀Apple 電腦的使用手册，以明瞭和電腦溝通，操作電腦的幾個基本程序，譬如，如何開機，如何進入Applesoft 作業系統，輸入程式，偵錯，以及執行程式等等的程序。
2. 選定一個章節閱讀第一節“目的”，看看此程式是否能引起您的興緻。若否，就試試下一個程式，直到引起您的興味。若您爲初學者，可先試試第六部份什錦程式中的簡短有趣程式。
3. 選好程式後，接著讀此章節的“使用方法”與“執行範例”，以深入了解此程式的執行細節。
4. 按入NEW 指令以清除Apple 電腦主記憶體內無用的程式。然後參照“程式”部分，細心地將程式打入電腦內。特別留意那些特殊符號(譬如，逗號，分號，冒號，問號等等)。
5. 完全將程式輸入後，使用LIST指令以顯示你剛剛輸入的程式，重複檢查是否有打字錯誤，刪行的現象。避免混淆了分號與冒號，或者將字母I與O看成1與0。
6. 在執行程式之前，先使用SAVE指令，將程式暫時儲存在磁碟或錄音帶內。如此可以預防一些意外事件(譬如，停電或機器故障)所造成的損失。

7. 按下RUN 執行程式。看看是否產生與“執行範例”相同的結果？若完全正確，恭喜您！然後跳到步驟 9。若發生錯誤，保持冷靜，然後按照步驟 8 處理。

8. 若電腦告訴您某行有 SYNTAX ERROR 邏輯錯誤，則按 LIST 指令然後詳細檢查那一行。也許你將字母拼錯，遺漏某字或將 0，1 看成字母 I 與 O。查出錯誤，更正後就跳回步驟 7。

若您得到別類的錯誤訊息，可參閱 Apple 使用手册，了解錯誤訊息的含義。有一概念必須謹記在心：錯誤也許並非發生在錯誤訊息指示的那一行，而往往是前一行方為罪魁禍首。另外一種可能情形是遺漏了某段或某一行。不管如何，修正錯誤後，然後跳回步驟 7。

若並沒出現任何錯誤訊息，但程式並沒按照“執行範例”產生預期的結果。這可能有兩種原因：首先，也許程式本身設計構想就是要使每次執行結果無法預期以避免重複（尤其是遊戲程式與畫圖程式），它們只是產生與“執行範例”同類的結果罷了！

第二種可能為您也許打錯了某行，其錯誤只是改變了程式的意義，並沒造成錯誤訊息。有一些小技巧可以找出此種錯誤，其原則就是逐步縮小可能發生錯誤的範圍。看看前面印出來的結果是否正確？若正確，錯誤大概就是位於印出此結果的PRINT指令之後方，依此概念逐步縮小範

圍就能找出錯誤。

9. 繼續執行程式，嘗試模仿出“執行範例”的結果。若你發現某些現象無法吻合“使用方法”所提的現象，就跳到步驟8。若一切無誤，就將程式SAVE儲存到磁碟或錄音帶上。

10. 閱讀“修改程式”部分。試試您所感興趣的一些修改方式。若您認爲修改後的版本較有趣，也可將其SAVE儲存在磁帶或錄音帶上。爲了避免混淆，可以修改程式前面的REM指令部分的文字說明。

目　錄

如何使用本書

第一部分　應用程式：家庭或工作的好幫手 …………………… 1

生命律 (BIORHY THM)

你能預測自己幸運的日子嗎……………… 3

支票簿 (CHECK BOOK)

平衡你的支票簿…………………………… 13

決斷 (DECIDE)

幫你做決策顧問…………………………… 25

借貸 (LOAN)

計算貸款的利息及償付方式……………… 39

耗油量 (MILEAGE)

分析你的車輛耗油率……………………… 48

問卷統計 (QUEST/EXAM)

幫你做問卷評閱及分析…………………… 58

第二部分　教育程式：幫助你學習 ……… 69

算術 (ARITHMETIC)

替小學生出算術練習題…………………… 71

記憶卡 (FLASHCARD)

建立記憶卡以資背誦 …………………… 81

公制 (METRIC)

學習公制系統 …………………………… 89

數字 (NUMBERS)

幫入學前兒童學習數字 ………………… 99

速讀機 (TACHIST)

使用速讀機增加你的閱讀速度 ………… 106

字彙 (VOCAB)

擴充你的字彙 …………………………… 116

第三部分　遊戲程式：和電腦一起玩耍 … 125

猜碼 (DECODE)

猜電腦的密碼 …………………………… 127

彩色部分 ………………………………… 132

呻吟的骰子 (GROAN)

可讓你呻吟的骰子戲 …………………… 145

猜字 (JOT)

向電腦挑戰猜字 ………………………… 159

障礙賽 (OBSTACLE)

跟你的朋友一起玩障礙賽跑 …………… 171

瘋狂大賽車 (ROADRACE)

經歷驚險的大賽車 ……………………… 180

瓦利棋 (WARI)

向電腦挑戰古非洲的瓦利棋 …………… 191

第四部分　圖形程式：視覺的享受 ……… 203

萬花筒 (KALEIDO)

八面對稱的萬花筒 ……………………… 204

星光閃爍 (SPARKLE)

閃爍耀眼的星星 ………………………… 209

同心圖 (SQUARES)

五彩同心方形 …………………………… 215

空中飛人 (WALLOONS)

蘋果馬戲團的空中飛人 ………………… 219

第五部分　數學程式：數學家、工程師與統計師的助手 ……………… 231

曲綫 (CURVE)

曲綫趨近 ………………………………… 233

微分方程式 (DIFFQN)

解微分方程式 …………………………… 246

圖表 (GRAPH)

顯示函數圖表 …………………………… 258

積分 (INTEGRATE)

決定曲綫內的面積 ……………………… 270

聯立方程式 (SIMEQN)

解綫性聯立方程式 ……………………… 279

統計 (STATS)

計算各統計量 …………………………… 287

第六部分　什錦程式：有趣的小程式……303

生日 (BIRTHDAY)

但願同年同月同日生……………………………305

圓周率 (PI)

計算π的近似值………………………………309

乘冪 (POWERS)

可到250位數的整數次方……………………316

直角三角形 (PYTHAG)

產生直角三角形（$a^2+b^2=c^2$）………322

附錄1　記憶體的使用……………………………329

參考書目…………………………………………331

第一部分　應用程式：

家庭或工作的好幫手

簡　介

個人電腦最主要的用途就是能實際應用到日常生活上的種種工作。Apple電腦是個多才多藝的好助手，以下六個程式可以幫助您在家庭、在商場上得心應手。

財務管理是個極重要的工作。 LOAN 借貸程式幫您計算房屋貸款、汽車貸款或商業貸款的利息及償付方式。您是否曾對支票簿的計帳平衡大感頭痛？CHECKBOOK 支票簿程式幫您檢查矯正每月的報帳，幫您找出裡面的錯誤。

駕駛人最關心的問題就是車輛的耗油量。MILEAGE 耗油量程式幫您追踪記錄您車輛的耗油量及使用效率。

若您常做問卷調查與統計的工作，常會對其煩不勝煩的問卷分析工作所苦惱。若您使用　QUEST/

EXAM 問卷統計程式，就可讓電腦幫您分憂分勞。實爲市場研究者、教師們的福音。

通常我們會面臨取捨不定的事情， DECIDE 決斷程式將Apple電腦轉變成可信賴的諮詢意見者。可幫您分析手邊的資料做一些正確的決定。

當您一早起床，也許想參考一下 BIORHYTHM 生命律程式，看看自己在今天的生理、心理週期的狀況。有些航空公司、工廠很重視生命律理論，若您也相信的話，可執此程式做爲行事的參考。

生命律(BIORHYTHM)

你能預測自己幸運的日子嗎？

目　的

您是否曾感覺到有些日子似乎不太對勁？我們每個人都會在某些日子顯得比平時笨拙、情緒低潮或者無法集中自己的注意力。有時候，我們知道其導因。它們或許來自於感冒感染或因為與好友吵架。然而，有時候我們找不到此類的理由。為何雖事事無誤，我們却沒來由地焦慮不安表現失常？

生命律定理（biorhythm theory）謂：每個人與生俱有生命週期，由出生的一刻起，隨著週期的起伏會影響我們的生理、情緒和智力狀態。在此不仔細推演生命律研究的來龍去脈（可自行到圖書館查心理學有關書籍），只是擇要指出生命律對我們所造成的影響。

生理週期為 23 天。在頭 11½ 天裡，您處於週期的正狀態，意味著您將會感到身體強健，精神飽滿

，健壯與耐力持久。在接著 11½ 天裡，您處於週期的負狀態，意味著耐力極差，易感到疲勞。

情緒週期爲 28 天。在頭 14 天裡，您處於週期的正狀態，意味著樂觀進取、和氣待人、心情愉快。在負狀態裡，則易感到情緒化、悲觀、脾氣暴燥。

第三種週期爲智力週期，一週期持續 33 天。在前半個週期裡，您將會感到創造力增強，易學新知識，智力活動頻繁。在負狀態時，您將會感到學習新觀念很困難，不如去複習舊知識。

這三個週期的上下起伏直接影響到每個個體。例如，假使您爲自制力極強，不受情緒影響的人，則您的情緒週期的高低起伏就不太明顯。同理，您的生理及智力的影響也依本身的身體狀況和智力能力而定。

此三個生命週期中，由正狀態轉變到負狀態（或由負狀態轉變到正狀態）的日子稱之爲“臨界日”。生命律定理指出，在生理週期或情緒週期的臨界日，個體有易生意外的傾向。智力週期的臨界日並不能視爲具危險性，但若與其他週期的臨界日相結合，則其潛在的危險即相對增加。我們可想見的，此三個週期的臨界日相重疊的時候，必須十分當心。

雖有此種說法，但生命律的理論尙有不少爭議。大部分科學家認爲目前並沒有足夠的證據來證實生命律所代表的意義。有些人相信生命週期確實存在，但並非像此處所提 23、28、33 天如此簡單一成不變的週期。

不管生命律的優、劣、眞、僞，這並非我們的主

題，此處的目的只是提出此有趣的定理，讓您可用 Apple 個人電腦加以處理，以達娛己娛人之效。

使用方法

程式首先問玩家的出生日期，以便計算生命週期。操作者輸入出生月、日，至於年，若爲 1900 到 1999 之間，則只須輸入後兩位數；若爲其他年代，則輸入四位數。

然後程式問操作者生命律圖表的開始顯示日期。此日期的輸入方法與前面相同，當然，此一日期不能先於出生日期。

大約一秒鐘的延遲後，程式自動清除螢光幕，以一天爲單位開始畫生命律圖。螢光幕的左邊顯示日期，右邊顯示圖表。圖表的左半部爲生命週期的負狀態，右半部爲正狀態。其中央綫爲臨界日，刻度爲零（非正狀態也非負狀態）。

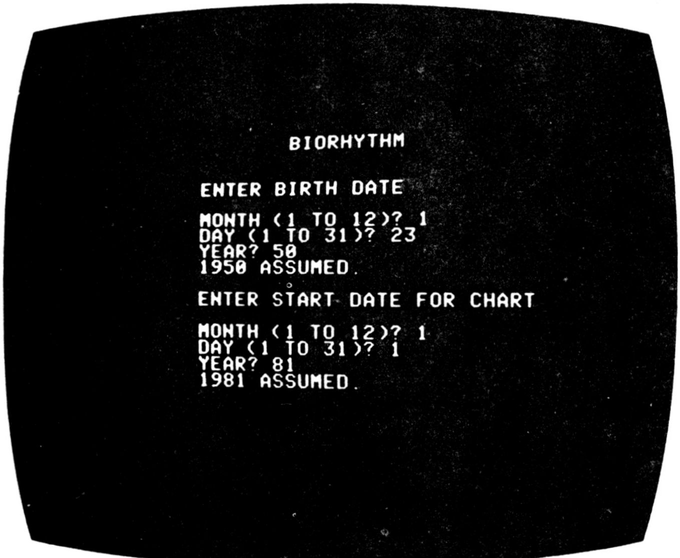
三個週期曲綫分別以字首字母加以區別——P代表生理週期（Physical），E代表情緒週期（Emotional），I代表智力週期（Intellectual）。當曲綫相重疊，則以星號代替字母。

螢光幕一次可顯示 18 天的生命律圖，然後程式等待使用者打入控制鍵。若打入E鍵，則停止目前圖表，程式從頭開始。若打入 SPACE 鍵（或RESET 與 SHIFT 除外的其他鍵），程式清除螢幕，顯示生

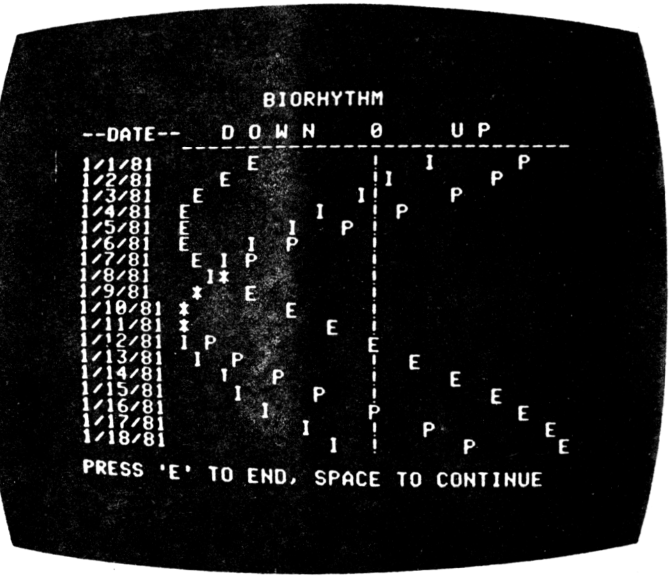
命律圖的下 18 天。若欲終止程式，則按 ESCAPE 鍵。

程式允許使用者輸入西元100年後的任何日期，而對於未來的日期並沒限制。但願此項限制並沒造成您的不便。

執行範例



操作者輸入其生日及生命律圖表開始顯示的日期。



程式顯示生命律圖表 18 天後，等待操作者按入一鍵以繼續顯示。

程　式

```
100 REM: BIORHYTHM
110 REM: COPYRIGHT 1980 BY TOM RUGG AND PHIL FELDMAN
120 L=0:Z=.99999:T=14:P=3.14159265
130 TEXT:HOME:PRINT TAB(8);"BIORHYTHM"
140 GOSUB 1200:PRINT:PRINT
150 PRINT"ENTER BIRTH DATE"
160 GOSUB 500
170 GOSUB 600
180 JB=JD
190 PRINT:PRINT"ENTER START DATE FOR CHART"
200 GOSUB 500
210 GOSUB 600
220 JC=JD
```

```
230 IF JC>=JB THEN 270
240 PRINT"CHART DATE CAN'T BE EARLIER"
250 PRINT"THAN BIRTH DATE.  TRY AGAIN."
260 GOTO 140
270 FOR J=1 TO 1000:NEXT
280 GOSUB 700
300 N=JC-JB
310 V=23:GOSUB 800:GOSUB 850
320 V=28:GOSUB 800:GOSUB 850
330 V=33:GOSUB 800:GOSUB 850
340 GOSUB 1000
350 PRINT TAB(1);C$;TAB(9);L$
360 JC=JC+1:L=L+1:IF L<18 THEN 300
370 PRINT:PRINT"PRESS 'E' TO END, SPACE TO CONTINUE";
380 GET R$:IF ASC(R$)=27 THEN END
390 IF R$="E" THEN 120
400 L=0:GOTO 280
500 PRINT
505 INPUT"MONTH (1 TO 12)? ";M
510 M=INT(M):IF M<1 OR M>12 THEN 505
520 INPUT"DAY (1 TO 31)? ";D
530 D=INT(D):IF D<1 OR D>31 THEN 520
540 INPUT"YEAR? ";Y
550 Y=INT(Y):IF Y<0 THEN 540
560 IF Y>99 THEN 580
570 Y=Y+1900:PRINT Y;" ASSUMED."
580 RETURN
600 W=INT((M-14)/12+Z)
610 JD=INT(1461*(Y+4800+W)/4)
620 B=367*(M-2-W*12)/12
630 IF B<0 THEN B=B+Z
640 B=INT(B):JD=JD+B
650 B=INT(INT(3*(Y+4900+W)/100)/4)
660 JD=JD+D-32075-B
670 RETURN
700 HOME
710 PRINT TAB(15);"BIORHYTHM"
720 PRINT:PRINT TAB(1);"--DATE--";TAB(12);
730 PRINT"D O W N";TAB(23);"0";TAB(29);"U P"
740 PRINT TAB(9);U$
750 RETURN
800 W=INT(N/V):R=N-W*V
810 RETURN
850 IF V<>23 THEN 960
```

```
855 L$=CHR$(32):FOR J=1 TO 4
860 L$=L$+L$:NEXT
870 L$=L$+LEFT$(L$,13)
880 L$=LEFT$(L$,T)+CHR$(33)+RIGHT$(L$,T)
890 IF V=23 THEN C$="P"
900 IF V=28 THEN C$="E"
910 IF V=33 THEN C$="I"
920 W=R/V:W=W*2*P
930 W=T*SIN(W):W=W+T+1.5
940 W=INT(W):A$=MID$(L$,W,1)
950 IF A$="P" OR A$="E" OR A$="*" THEN C$="*"
955 IF W=1 THEN 980
957 IF W=T+T+1 THEN 990
960 L$=LEFT$(L$,W-1)+C$+RIGHT$(L$,T+T+1-W)
970 RETURN
980 L$=C$+RIGHT$(L$,T+T):RETURN
990 L$=LEFT$(L$,T+T)+C$:RETURN
1000 W=JC+68569:R=INT(4*W/146097)
1010 W=W-INT((146097*R+3)/4)
1020 Y=INT(4000*(W+1)/1461001)
1030 W=W-INT(1461*Y/4)+31
1040 M=INT(80*W/2447)
1050 D=W-INT(2447*M/80)
1060 W=INT(M/11):M=M+2-12*W
1070 Y=100*(R-49)+Y+W
1080 A$=STR$(M)
1090 C$=A$+"/"
1100 A$=STR$(D)
1110 C$=C$+A$+"/"
1120 A$=STR$(Y):W=LEN(A$)
1130 C$=C$+MID$(A$,W-1,2)
1140 RETURN
1200 U$="-":FOR J=1 TO 4
1210 U$=U$+U$:NEXT
1220 U$=U$+LEFT$(U$,13)
1230 RETURN
```

修改程式

1. 若欲看任兩個日期間的日差數，可插入此行：

```
305 PRINT "DAYS=";N:END
```

然後輸入第一個日期做爲出生日期，第二個日期爲生命律圖的起始顯示日期，則可使程式顯示此二日期的日數差。

2. 欲改變螢幕每次顯示的圖表日數，可修改360行的 18 。

主流程

120 － 140	設立變數的初期狀態，顯示標題。
150 － 180	詢問出生日期，然後將其轉換成朱麗安曆型式（ Julian Date Format ）（註：由公元前 4713 年一月一日開始算的日子個數）。
190 － 220	詢問生命律圖的開始顯示日期，然後將其轉換成朱麗安曆型式。
230 － 260	檢查圖表日期不早於出生日期。
270	圖表顯像前，先延遲一秒鐘。
280	顯像圖表的標題。
300	計算出生日期與目前圖表日期之間的日差數。
310 － 330	對三種生命週期填入其適當位置於 L$ 字串內。
340	將朱麗安（ Julian ）日期型式轉換回月

-日-年格式。

350	顯像圖表內的一行。
360－400	圖表日期加1，檢查是否螢幕已滿。
500－580	詢問月-日-年的副程式，且負責回答的管制。
600－670	副程式，專司月-日-年格式轉換爲朱麗安格式。
700－750	副程式，清除螢幕與標題顯像。
800－810	副程式，計算N/V的餘數R。
850－990	副程式，依V與R畫L$內的一點。
1000－1140	副程式，將朱麗安日期 JC 轉換回月/日/年格式。
1200－1230	副程式，建立U$。

主要變數

L	計數螢幕顯示行數。
T	圖表由中間算起，半邊的字元數目。
P	π
JB	出生日期的朱麗安格式
JD	副程式計算的朱麗安日期
JC	圖表開始日期的朱麗安日期
K	迴圈與工作變數
N	出生日期與目前圖表日期的日數差
V	目前生命週期的日數（23、28或33）

C$	月/日/年格式的日期字串
L$	生命律圖一行的字串
R$	螢幕填滿後，使用者的回答
M	月（1－12）
D	日（1－31）
Y	年（100或更大）
W、B	工作變數
R	N/V 的餘數（註：週期內的日數）
A$	工作變數
U$	虛綫字串

研究建議

調查歷史著名的人物或運動家的生命律現象。例如，田徑運動員是否都在生理週期的正狀態之時改寫世界記錄？是否林肯在其情緒與智力週期高潮中撰寫了“蓋茨堡演講稿”？是否有極高比率的意外事件發生於人們的臨界日？

支票簿(CHECK BOOK)

平衡你的支票簿

目 的

很多人將每月的支票簿收支平衡計算視爲惱人易錯的工作。有些人於首次嚐試就搞混而只好放棄；而有些人則好不容易算出後，却無法與銀行的說法符合而放棄。現在，您握有一支生力軍——個人電腦。此程式教您如何操作平衡計算支票簿的工作，並做一切必須的運算。

使用方法

程式首先印出一些指示，告訴您如何檢查每張簽付支票金額與存款金額是否與支票簿所記載的符合。有時候，銀行出納員可能將您書寫在支票上的金額讀錯（尤其是字跡潦草之時）或將金額抄錯。當您逐項

檢查支票簿時，可於每項支票與存款條列上加上核對記號。一個優良系統應爲每個月有不同的核對記號（也許這一月爲“ x ”符號，下個月則爲另一個符號），如此可方便用戶對照出何項支票與存款金額的來源。

然後程式詢問銀行結算的平衡金額。接著詢問銀行所記載您最近開出的支票號碼（並非金額）。此號碼通常爲銀行目前處理的最高支票號碼，除非您不按順序開支票。您所計算的會計平衡必須於此最近支票項後與銀行記載的平衡相符合，此即程式接著所要詢問的——最近的支票後之支票簿平衡金額。

程式必須補償您的支票簿在最近所開的支票之前所記載金額與銀行記載金額的差額。首先，若您在最近支票前，曾有存款而銀行記載上却缺少，則必須輸入這些存款。一般而言都無此情形，因此您只要打入“ END ”。

接著您必須輸入在目前最近支票以前尙未被銀行結算的任何支票金額。檢查您的支票簿內沒標上核對記號的支票項目。留意這些項目可能爲數月以前遺漏下來的。

然後您打入銀行報告內記載而尙未歸入支票簿的服務費用或負債項目。通常，這些支出款項爲每月的服務費用，或者印新支票的額外費用或其他雜費支出。存款項目在此時暫不要輸入。必須確使這些支出雜費轉載到您的支票簿，方能使其與下個月的銀行報告相平衡。

最後，要求您輸入銀行報告記載而尙未歸入支票

簿的存款或貸款項目。這些存款項目不止一項並不足為奇，因為銀行通常優先處理存款事物而將支票的支付挪後。

現在就到揭曉的時候了！程式宣佈您的支票簿是否平衡，然後顯示總和。若果真平衡，則將結果儲存起來，等待下個月的銀行報告寄到。

若不平衡，您必須找出何處出錯。程式提供七個選擇鍵，讓您重溫前面輸入的項目以防打字錯誤。若找出錯誤，則重頭打一遍。當然，若導致不平衡的只是一個簡單的錯誤，則不須要完全從頭重覆一遍。

若您輸入的所有資料都正確，不平衡的最大可能原因為您手上支票簿的計算錯誤。特別檢查您的加法與減法，尤其減法往往是罪魁禍首。另外一個常犯的錯誤為將減法誤算為加法。若為此類錯誤，您的錯誤差額將為開出支票面額的兩倍，因此很容易偵查出錯誤所在。

假如以上原因尚無法解釋您的錯誤，檢查支票簿於平衡計算時，是否已減去上月的服務費用。若您的支票簿於上個月已沒平衡，自然無法期待在本月的平衡計算中得到平衡。

程式對您輸入的每類資料（譬如：未付清的支票存款等等）的項目個數都有限制。但此項限制很容易修改。見後面的“修改程式”部份。

註：見書前的序。

執行範例

```
                CHECKBOOK BALANCER

FIRST, COMPARE THE BANK STATEMENT
WITH YOUR CHECKBOOK.
MAKE SURE THE STATEMENT AND THE
CHECKBOOK SHOW THE SAME FIGURES
FOR EACH CHECK AND DEPOSIT.

MAKE A MARK IN THE CHECKBOOK NEXT TO
EACH CHECK AND DEPOSIT LISTED
ON THE STATEMENT.

WHAT'S THE ENDING BALANCE SHOWN
ON THE STATEMENT?
?520.16

NOW FIND THE MOST RECENT CHECK THAT
IS SHOWN ON THE BANK STATEMENT.
WHAT IS THE CHECK NUMBER OF THIS CHECK?
?1652

WHAT BALANCE DOES YOUR CHECKBOOK
SHOW AFTER CHECK NO. 1652
?480.12

ENTER THE AMOUNT OF EACH DEPOSIT
THAT IS SHOWN IN YOUR CHECKBOOK
PRIOR TO CHECK NO. 1652
BUT IS NOT ON THE STATEMENT.
WHEN NO MORE, SAY 'END'
?END

TOTAL = 0

NOW ENTER THE AMOUNTS OF ANY CHECKS
THAT ARE IN YOUR CHECKBOOK PRIOR
TO CHECK 1652 BUT THAT
HAVE NOT BEEN SHOWN ON A BANK
STATEMENT YET.
```

```
WHEN NO MORE, SAY 'END'
?35.04
?10
?END

TOTAL = 45.04

NOW ENTER THE AMOUNTS OF ANY
SERVICE CHARGES OR DEBIT MEMOS.

WHEN NO MORE, SAY 'END'
?2.35
?2.65
?END

TOTAL = 5

ENTER THE AMOUNT OF EACH DEPOSIT
THAT IS SHOWN IN YOUR CHECKBOOK
AFTER CHECK NO. 1652 THAT IS
ALSO LISTED ON THE STATEMENT.

WHEN NO MORE, SAY 'END'
?END

TOTAL = 0

CONGRATULATIONS! IT BALANCES.

STATEMENT BALANCE + CHECKS OUTSTANDING
+ SERVICE CHARGES = 525.16

CHECKBOOK BALANCE + CHECKS OUTSTANDING
+ RECENT DEPOSITS = 525.16

DIFFERENCE = 0

PRESS ANY KEY TO CONTINUE

NEXT ACTION:
 1 - LIST CHECKS OUTSTANDING
 2 - LIST DEPOSITS OUTSTANDING
 3 - LIST SERVICE CHARGES
 4 - START OVER
 5 - END PROGRAM
```

```
 6 - DISPLAY BALANCING INFO
 7 - LIST DEPOSITS AFTER LAST CHECK
?5
```

程 式

```
100 REM: CHECKBOOK
110 REM: COPYRIGHT 1980 BY TOM RUGG AND PHIL FELDMAN
120 TEXT:HOME
130 PRINT"CHECKBOOK ANALYZER"
140 PRINT:PRINT
150 MC=20:MD=10:MS=10:MR=10
155 L$="NO MORE ROOM"
160 DIM C(MC),D(MD),S(MS),R(MR)
170 TC=0:TD=0:TS=0:TR=0:NC=0:ND=0:NS=0:NR=0
180 E$="ERROR. RE-ENTER, PLEASE."
190 PRINT"FIRST, COMPARE THE BANK STATEMENT"
200 PRINT"WITH YOUR CHECKBOOK."
210 PRINT
220 PRINT"MAKE SURE THE STATEMENT AND THE"
230 PRINT"CHECKBOOK SHOW THE SAME FIGURES"
240 PRINT"FOR EACH CHECK AND DEPOSIT."
250 PRINT:PRINT"MAKE A MARK IN THE CHECKBOOK NEXT TO"
260 PRINT"EACH CHECK AND DEPOSIT LISTED"
270 PRINT"ON THE STATEMENT."
280 PRINT:PRINT"WHAT'S THE ENDING BALANCE SHOWN"
290 PRINT"ON THE STATEMENT?":INPUT SB
300 PRINT:PRINT"NOW FIND THE MOST RECENT CHECK THAT"
310 PRINT"IS SHOWN ON THE BANK STATEMENT."
320 PRINT
330 PRINT"WHAT IS THE CHECK NUMBER OF"
340 PRINT"THIS CHECK?":INPUT LC
350 IF LC=INT(LC) THEN 380
360 PRINT"NO, NOT THE AMOUNT OF THE CHECK."
370 GOTO 300
380 PRINT
390 PRINT"WHAT BALANCE DOES YOUR CHECKBOOK"
400 PRINT"SHOW AFTER CHECK NO. ";LC
410 INPUT CB
```

```
420 PRINT:PRINT
430 PRINT"ENTER THE AMOUNT OF EACH DEPOSIT"
440 PRINT"THAT IS SHOWN IN YOUR CHECKBOOK"
450 PRINT"PRIOR TO CHECK NO. ";LC
460 PRINT"BUT IS NOT ON THE STATEMENT."
470 A$="WHEN NO MORE, SAY 'END'":PRINT:PRINT A$
480 INPUT R$
490 IF R$="END" THEN 545
500 IF VAL(R$)>0 THEN 520
510 PRINT:PRINT E$:GOTO 470
520 ND=ND+1:D(ND)=VAL(R$):TD=TD+D(ND)
530 IF ND<MD THEN 480
540 PRINT:PRINT L$
545 PRINT:PRINT"TOTAL = ";TD:PRINT
550 PRINT"NOW ENTER THE AMOUNTS OF ANY CHECKS"
560 PRINT"THAT ARE IN THE CHECKBOOK PRIOR"
570 PRINT"TO CHECK ";LC;" BUT THAT"
580 PRINT"HAVE NOT BEEN SHOWN ON A BANK"
590 PRINT"STATEMENT YET."
600 PRINT:PRINT A$
610 INPUT R$
620 IF R$="END" THEN 690
630 IF VAL(R$)>0 THEN 660
640 PRINT:PRINT E$
650 GOTO 600
660 NC=NC+1:C(NC)=VAL(R$):TC=TC+C(NC)
670 IF NC<MC THEN 610
680 PRINT:PRINT L$
690 PRINT:PRINT"TOTAL = ";TC:PRINT
700 PRINT"NOW ENTER THE AMOUNTS OF ANY"
710 PRINT"SERVICE CHARGES OR DEBIT MEMOS."
720 PRINT:PRINT A$
730 INPUT R$
740 IF R$="END" THEN 800
750 IF VAL(R$)>0 THEN 770
760 PRINT:PRINT E$:GOTO 720
770 NS=NS+1:S(NS)=VAL(R$):TS=TS+S(NS)
780 IF NS<MS THEN 730
790 PRINT:PRINT L$
800 PRINT:PRINT"TOTAL = ";TS:PRINT
805 GOSUB 2000
810 W=SB+TD+TS-CB-TC-TR:W=ABS(W)
815 IF W<.001 THEN W=0
817 IF W<>0 THEN 840
```

```
820 PRINT:PRINT"CONGRATULATIONS!  IT BALANCES."
830 GOTO 850
840 PRINT:PRINT"SORRY, ITS OUT OF BALANCE."
850 PRINT
860 PRINT"STATEMENT BALANCE + DEPOSITS OUTSTANDING"
870 PRINT"+ SERVICE CHARGES = ";SB+TD+TS
880 PRINT
890 PRINT"CHECKBOOK BALANCE + CHECKS OUTSTANDING"
900 PRINT"+ RECENT DEPOSITS = ";CB+TC+TR
910 PRINT
920 PRINT"DIFFERENCE = ";W
925 PRINT:PRINT"PRESS ANY KEY TO CONTINUE"
930 GET R$
935 PRINT
940 PRINT"NEXT ACTION:"
950 PRINT" 1 - LIST CHECKS OUTSTANDING"
960 PRINT" 2 - LIST DEPOSITS OUTSTANDING"
970 PRINT" 3 - LIST SERVICE CHARGES"
980 PRINT" 4 - START OVER"
990 PRINT" 5 - END PROGRAM"
1000 PRINT" 6 - DISPLAY BALANCING INFO"
1010 PRINT" 7 - LIST DEPOSITS AFTER LAST CHECK"
1020 INPUT R$:R=VAL(R$)
1030 IF R<1 OR R>7 THEN 1050
1040 ON R GOTO 1100,1200,1300,1400,1500,850,1700
1050 PRINT:PRINT E$:GOTO 935
1100 PRINT:PRINT"CHECKS OUTSTANDING"
1110 FOR J=1 TO NC
1120 PRINT C(J):NEXT
1130 GOTO 925
1200 PRINT:PRINT"DEPOSITS OUTSTANDING"
1210 FOR J=1 TO ND
1220 PRINT D(J):NEXT
1230 GOTO 925
1300 PRINT:PRINT"SERVICE CHARGES"
1310 FOR J=1 TO NS
1320 PRINT S(J):NEXT
1330 GOTO 925
1400 CLEAR:GOTO 120
1500 END
1700 PRINT:PRINT"RECENT DEPOSITS"
1710 FOR J=1 TO NR
1720 PRINT R(J):NEXT
1730 GOTO 925
```

```
2000 PRINT
2010 PRINT"ENTER THE AMOUNT OF EACH DEPOSIT"
2020 PRINT"THAT IS SHOWN IN YOUR CHECKBOOK"
2030 PRINT"AFTER CHECK NO. ";LC;" THAT IS"
2040 PRINT"ALSO LISTED IN THE STATEMENT."
2050 PRINT:PRINT A$
2060 INPUT R$
2070 IF R$="END" THEN 2130
2080 IF VAL(R$)>0 THEN 2100
2090 PRINT:PRINT E$:GOTO 2050
2100 NR=NR+1:R(NR)=VAL(R$):TR=TR+R(NR)
2110 IF NR<MR THEN 2060
2120 PRINT:PRINT L$
2130 PRINT:PRINT"TOTAL = ";TR:PRINT
2140 RETURN
```

修改程式

更改程式中對每類輸入項目的個數限制。在150行建立這些限制。例如，假如您有時候的未付清支票超過 20 項，可更改程式150行的MC 值爲 50 。若您預期的項目超過 10 項，也可更改其他三個變數。它們分別爲： MD ——未入帳的存款， MS ——服務費用或負債項目個數，以及 MR ——目前存款與貸款個數。若欲做這些修改，電腦內必須多保留 4100 byte 的記憶空間。（見附錄 1 ）

主流程

120 － 290　設立變數以及顯示第一個指示。

300－370	輸入目前最近支票號碼。
380－410	輸入目前支票號碼後的平衡值。
420－545	輸入未入帳的存款。
550－690	輸入未付清的支票。
700－800	輸入服務費用與負債項目。
805	輸入目前存款與貸款項目。
810－920	處理平衡計算，顯示計算結果。
925－1050	詢問下個步驟。跳到適當的副程式。
1100－1130	副程式，顯示未付清的支票。
1200－1230	副程式，顯示未入帳的存款。
1300－1330	副程式，顯示服務費用與負債項目。
1400	重新執行程式。
1500	結束程式。
1700－1730	副程式，顯示目前存款。
2000－2140	副程式，輸入目前存款。

主要變數

MC	未付清支票項目的最大個數
MD	未入帳存款項目的最大個數
MS	服務費用、負債項目的最大個數
MR	目前存款、貸款項目的最大個數
C	未付清支票陣列
D	未入帳存款陣列
S	服務費用與負債項目陣列

R	目前存款與貸款項目陣列
TC	未付清支票總和
TD	未入帳存款總和
TS	服務費用與負債項目總和
TR	目前存款與貸款項目總和
NC	未付清支票個數·
ND	未入帳存款個數
NS	服務費用與負債項目總和
NR	目前存款與貸款項目總和
E$	錯誤訊息
SB	銀行報告平衡值
LC	報告上的最後支票號碼
CB	報告上最後支票後的支票簿平衡值
R$	操作員的回答
W	支票簿平衡值的差額
R	下一個步驟的回答數值
A$	指示如何表達已沒有資料了。
L$	表示無多餘記憶空間給資料的訊息
J	迴圈變數

研究建議

1. 添加較完整的操作與介紹訊息，讓此程式能教導那些從未處理支票簿平衡計算的新手。
2. 允許操作員更改已發現輸入錯誤的項目。欲達成

此項功能，可額外加入一個選擇項於“NEXT ACTION”列表，此選擇項詢問操作員欲更改的輸入類別。此項功能可促使操作員不須從頭輸入資料，即能修改錯誤項目。

3. 若支票簿無法達到平衡，讓程式做一些分析（就如“使用方法”中所提議的法則），然後指出最可能導致此錯誤狀況的原因。

4. 幫助操作者尋找支票簿上的計算錯誤。先詢問初始平衡值，然後詢問每項支票或存款的金額。依操作者的指示做加法或減法。每個項目後，即顯示新的平衡值，使操作者可逐項比較支票簿各項。

決斷(DECIDE)

幫你做決策顧問

目 的

“抉擇！抉擇！”有多少次當您面臨取捨舉棋不定之時，會吐露此句話？是否覺得有很多場合，一個可資信賴的建議者有極大的助益？那麼，現在您有一個相當好的幫手——Apple 個人電腦。

此程式可幫助您由多個選擇中挑選一個最佳的決斷。它首先由您身上收集相關的資料，然後依合理數量化的方式加以組織，然後指出最佳決定，以及其他可能選擇的相關取捨比率。

您可使用此程式做各種類型的決斷。其適用的範圍譬如選擇最適當的立體音響設備、職業工作的取捨、商業交易的決策等等。每一項決斷都與您個人的決定息息相關。

使用方法

程式首先要求您將手邊欲處理的決斷區分爲三類，其分別爲：

1) 選擇一項（或一件物品）
2) 選擇一種行爲動作
3) 做是或非的決定

您可簡單地按1、2或3，然後按RETURN鍵來指示您目前面臨的決斷種類。若您按入1，意味著欲選擇一項物品，程式就會詢問您是何種物品。

若爲頭兩種決斷，您必須接著輸入列入考慮的所有可能情形。程式以問號“？”來催促您輸入每一項候選情形。當輸入所有可能情形後，則打入“END”字於催促問號後。當然，您輸入的可能情形一定至少有兩項（我們不希望您面臨單一可能情形下，還會發生決斷的困擾）。可能情形列表完全輸入後，程式自動顯示一遍，讓您核定無誤。若有錯誤，可重新輸入此列表。

現在您必須整理出影響您決策行爲的重要考慮因素。例如：位置、價錢與環境設備是您租房子的重要考慮因素。若欲購買電冰箱，您的考慮因素可能爲價錢、容量、可靠度、以及售後服務。不管何種決策例子，程式都以問號來催促您輸入考慮因素，一個問號輸入一個考慮因素，然後以“END”字來結束此考慮

因素列表。完成輸入後，程式重新顯示此列表。然後您必須決定那一因素爲最關鍵因素，輸入此因素的號碼。（若您想改變考慮因素列表的內容，可按入 0 ）

程式接著要求您以最關鍵因素爲準，安排其他考慮因素的相對重要性比值。安排方法爲首先設關鍵因素爲 10 ，然後依其重要性設其他因素爲 0 到 10 之間的任意數目。這些數目的大小直接反映出每個考慮因素的重要性， 10 分爲最重要，較小值其重要性也跟著遞減。

現在您必須安排決斷可能情形相對於每項考慮因素的比值。每項考慮因素必須獨立處理。針對於一項考慮因素時，您必須設定每個決斷可能情形滿足此項因素的比值。程式首先訂一樣決斷可能情形的比值爲 10 ，然後您依別項決斷可能情形相對於它滿足此因素的狀況，設定其他決斷可能情形的比值（高於、等於、或低於 10 ）。

舉例說明可能更易於了解。假定您想養一寵物，有狗、貓與金絲雀可資選擇。此時，喜愛程度爲您最主要考慮因素。若只考慮喜愛因素，程式設貓的值爲 10 ，然後您也許設狗的值爲 20 、金絲雀爲 6.5 。此意味著您愛狗的程度爲貓的兩倍，而對金絲雀的好感只爲貓的三分之二。

具有此項資料後，程式即能決定何種選擇最適合您，且可依序排出決斷所有可能情形的先後順序。每一項都有一個比值大小，其最佳的選擇爲100 分。

或許此決斷程式無法當做優秀、有條理的思考能

力之取代品。然而，它通常也能提供一點有價值的洞察力。您也許會發現有些結果的得分出奇地高或出奇地低，但若改進輸入的資料後重新執行程式，即能獲得較明確的答案。至少，它可幫助您以系統化誠實的眼光來做決斷。

執行範例

```
                    DECIDE

   I CAN HELP YOU MAKE A DECISION.  ALL
I NEED TO DO IS ASK SOME QUESTIONS AND
THEN ANALYZE THE INFORMATION YOU GIVE.

     ----------------------------

WHICH OF THESE BEST DESCRIBES THE TYPE
OF DECISION FACING YOU?

   1) CHOOSING AN ITEM FROM VARIOUS
      ALTERNATIVES.

   2) CHOOSING A COURSE OF ACTION FROM
      VARIOUS ALTERNATIVES.

   3) MAKING A 'YES' OR 'NO' DECISION.

WHICH ONE (1, 2, OR 3)? 1

WHAT TYPE OF ITEM MUST YOU DECIDE UPON
? VACATION

   I NEED TO HAVE A LIST OF EACH
VACATION UNDER CONSIDERATION.

   INPUT THEM ONE AT A TIME
IN RESPONSE TO EACH QUESTION MARK.
```

```
   THE ORDER IN WHICH YOU INPUT THEM
HAS NO PARTICULAR SIGNIFICANCE.

   TYPE THE WORD 'END' TO INDICATE
THAT THE WHOLE LIST HAS BEEN ENTERED.

? MOUNTAIN CAMPING
? AFRICAN SAFARI
? TRIP TO WASHINGTON D.C.
? END

O.K. HERE'S THE LIST YOU'VE GIVEN ME:

  1)  MOUNTAIN CAMPING
  2)  AFRICAN SAFARI
  3)  TRIP TO WASHINGTON D.C.

IS THIS LIST CORRECT (Y OR N) ? YES

   NOW, THINK OF THE DIFFERENT FACTORS
THAT ARE IMPORTANT TO YOU IN CHOOSING
THE BEST VACATION.

   INPUT THEM ONE AT A TIME IN RESPONSE
TO EACH QUESTION MARK.

   TYPE THE WORD 'END' TO TERMINATE
THE LIST.

? RELAXATION
? AFFORDABILITY
? CHANGE OF PACE
? END

HERE'S THE LIST OF FACTORS YOU GAVE ME:

  1)  RELAXATION
  2)  AFFORDABILITY
  3)  CHANGE OF PACE
   DECIDE WHICH FACTOR ON THE LIST IS
THE MOST IMPORTANT AND INPUT ITS NUMBER.
(TYPE 0 IF THE LIST NEEDS CHANGING.)

? 2
```

```
   NOW LET'S SUPPOSE WE HAVE A SCALE OF
IMPORTANCE RANGING FROM 0-10.

   WE'LL GIVE AFFORDABILITY A
VALUE OF 10 SINCE AFFORDABILITY
WAS RATED MOST IMPORTANT.

   ON THIS SCALE, WHAT VALUE OF
IMPORTANCE WOULD THE OTHER FACTORS HAVE?

RELAXATION
? 5.5

CHANGE OF PACE
? 9

   EACH VACATION
MUST NOW BE COMPARED WITH RESPECT TO
EACH IMPORTANCE FACTOR.

   WE'LL CONSIDER EACH FACTOR
SEPARATELY AND THEN RATE
EACH VACATION IN TERMS
OF THAT FACTOR ONLY.

   LET'S GIVE MOUNTAIN CAMPING
A VALUE OF 10 ON EVERY SCALE.

   THEN EVERY OTHER VACATION
WILL BE ASSIGNED A VALUE HIGHER OR
LOWER THAN 10.  THIS VALUE DEPENDS ON
HOW MUCH YOU THINK IT IS BETTER OR
WORSE THAN MOUNTAIN CAMPING.
   -----------------
   CONSIDERING ONLY RELAXATION AND
ASSIGNING 10 TO MOUNTAIN CAMPING ;
WHAT VALUE WOULD YOU ASSIGN TO

AFRICAN SAFARI? 3

TRIP TO WASHINGTON D.C.? 9

   -----------------
   CONSIDERING ONLY AFFORDABILITY AND
ASSIGNING 10 TO MOUNTAIN CAMPING ;
```

```
WHAT VALUE WOULD YOU ASSIGN TO

AFRICAN SAFARI? 1

TRIP TO WASHINGTON D.C.? 8

   -----------------
   CONSIDERING ONLY CHANGE OF PACE AND
ASSIGNING 10 TO MOUNTAIN CAMPING ;
WHAT VALUE WOULD YOU ASSIGN TO

AFRICAN SAFARI? 60

TRIP TO WASHINGTON D.C.? 25

TRIP TO WASHINGTON D.C. COMES OUT BEST
BUT IT'S VERY CLOSE.

HERE IS THE FINAL LIST IN ORDER.

TRIP TO WASHINGTON D.C. HAS BEEN
GIVEN A VALUE OF 100 AND THE OTHERS
RATED ACCORDINGLY.

   --------------------

100             TRIP TO WASHINGTON D.C.
98.6587184      MOUNTAIN CAMPING
78.8375559      AFRICAN SAFARI
```

程 式

```
100 REM: DECIDE
110 REM: COPYRIGHT 1980 BY PHIL FELDMAN AND TOM RUGG
150 TEXT
160 DIM L$(10),F$(10),V(10),C(10,10),D(10),Z(10)
180 E$="END"
200 GOSUB 2000
210 PRINT"   I CAN HELP YOU MAKE A DECISION.  ALL"
220 PRINT"I NEED TO DO IS ASK SOME QUESTIONS AND"
```

```
230 PRINT"THEN ANALYZE THE INFORMATION YOU GIVE.":PRINT
240 PRINT"     ----------------------------":PRINT
250 PRINT"WHICH OF THESE BEST DESCRIBES THE TYPE"
260 PRINT"OF DECISION FACING YOU?":PRINT
270 PRINT"  1) CHOOSING AN ITEM FROM VARIOUS"
280 PRINT"     ALTERNATIVES.":PRINT
290 PRINT"  2) CHOOSING A COURSE OF ACTION FROM"
300 PRINT"     VARIOUS ALTERNATIVES.":PRINT
310 PRINT"  3) MAKING A 'YES' OR 'NO' DECISION.":PRINT
320 PRINT"WHICH ONE (1, 2, OR 3)";
330 INPUT"? ";R$
340 T=INT(VAL(R$)):IF T<1 OR T>3 THEN 320
350 GOSUB 2000
400 ON T GOTO 410,440,470
410 PRINT"WHAT TYPE OF ITEM MUST YOU DECIDE UPON"
420 INPUT"? ";T$:GOTO 500
440 T$="COURSE OF ACTION":GOTO 500
470 T$="'YES' OR 'NO'"
480 NI=2:L$(1)="DECIDING YES":L$(2)="DECIDING NO"
490 GOTO 750
500 GOSUB 2000:NI=0
510 PRINT"   I NEED TO HAVE A LIST OF EACH"
520 PRINT T$;" UNDER CONSIDERATION.":PRINT
530 PRINT"   INPUT THEM ONE AT A TIME"
540 PRINT"IN RESPONSE TO EACH QUESTION MARK.":PRINT
550 PRINT"   THE ORDER IN WHICH YOU INPUT THEM"
560 PRINT"HAS NO PARTICLULAR SIGNIFICANCE.":PRINT
570 PRINT"   TYPE THE WORD '";E$;"' TO INDICATE"
580 PRINT"THAT THE WHOLE LIST HAS BEEN ENTERED.":PRINT
590 NI=NI+1:INPUT"? ";L$(NI)
600 IF L$(NI)<>E$ THEN 590
610 NI=NI-1
620 IF NI>=2 THEN 650
630 PRINT:PRINT"YOU MUST HAVE AT LEAST 2 CHOICES":PRINT
640 PRINT"TRY AGAIN":GOSUB 2100:GOTO 500
650 GOSUB 2000:PRINT
    "O.K. HERE'S THE LIST YOU'VE GIVEN ME:":PRINT
660 FOR J=1 TO NI:PRINT"   ";J;") ";L$(J):NEXT:PRINT
670 PRINT"IS THIS LIST CORRECT (Y OR N) ? ";
680 GET R$
690 IF R$="Y" THEN PRINT"YES":GOTO 750
700 IF R$="N" THEN PRINT"NO"
710 IF R$="N" THEN PRINT:PRINT
    "THE LIST MUST BE RE-ENTERED"
720 IF R$="N" THEN GOSUB 2100:GOSUB 500
```

```
730 GOTO 680
750 GOSUB 2000
760 PRINT"   NOW, THINK OF THE DIFFERENT FACTORS"
770 IF T<3 THEN PRINT
    "THAT ARE IMPORTANT TO YOU IN CHOOSING"
780 IF T<3 THEN PRINT"THE BEST ";T$;"."
790 IF T=3 THEN PRINT
    "THAT ARE IMPORTANT TO YOU IN DECIDING":PRINT
    "YES OR NO"
800 PRINT:PRINT"   INPUT THEM ONE AT A TIME IN RESPONSE"
810 PRINT"TO EACH QUESTION MARK.":PRINT
820 PRINT"   TYPE THE WORD '";E$;"' TO TERMINATE"
830 PRINT"THE LIST.":PRINT:NF=0
840 NF=NF+1:INPUT"? ";F$(NF)
850 IF F$(NF)<>E$ THEN 840
860 NF=NF-1:PRINT
870 IF NF<1 THEN PRINT
    "YOU MUST HAVE AT LEAST ONE! - REDO IT"
880 IF NF<1 THEN GOSUB 2100:GOTO 750
890 GOSUB 2000:PRINT
    "HERE'S THE LIST OF FACTORS YOU GAVE ME:":PRINT
900 FOR J=1 TO NF:PRINT"    ";J;") ";F$(J):NEXT:PRINT
910 PRINT"   DECIDE WHICH FACTOR ON THE LIST IS"
920 PRINT"THE MOST IMPORTANT AND INPUT ITS NUMBER."
930 PRINT"(TYPE 0 IF THE LIST NEEDS CHANGING.)":PRINT
940 INPUT"? ";A:A=INT(A):IF A=0 THEN 750
950 IF A<0 OR A>NF THEN 890
1000 GOSUB 2000:IF NF=1 THEN 1200
1010 PRINT"   NOW LET'S SUPPOSE WE HAVE A SCALE OF"
1020 PRINT"IMPORTANCE RANGING FROM 0-10.":PRINT
1030 PRINT"   WE'LL GIVE ";F$(A);" A"
1040 PRINT"VALUE OF 10 SINCE ";F$(A)
1050 PRINT"WAS RATED THE MOST IMPORTANT.":PRINT
1060 PRINT"   ON THIS SCALE, WHAT VALUE OF"
1070 PRINT"IMPORTANCE WOULD THE OTHER FACTORS HAVE?"
1080 FOR J=1 TO NF:Q=A:IF J=Q THEN 1110
1090 PRINT:PRINT F$(J):INPUT"? ";V(J)
1100 IF V(J)<0 OR V(J)>10 THEN PRINT
    " IMPOSSIBLE VALUE - TRY AGAIN":GOTO 1090
1110 NEXT
1200 V(A)=10:Q=0:FOR J=1 TO NF:Q=Q+V(J):NEXT:FOR J=1 TO
    NF
1210 V(J)=V(J)/Q:NEXT:GOSUB 2000
1220 IF T<>3 THEN PRINT"   EACH ";T$
1230 IF T=3 THEN PRINT"   DECIDING YES OR DECIDING NO"
```

```
1240 PRINT"MUST NOW BE COMPARED WITH RESPECT TO"
1250 PRINT"EACH IMPORTANCE FACTOR.":PRINT
1260 PRINT"   WE'LL CONSIDER EACH FACTOR"
1270 PRINT"SEPARATELY AND THEN RATE"
1280 IF T<>3 THEN PRINT"EACH ";T$;" IN TERMS"
1290 IF T=3 THEN PRINT
     "DECIDING YES OR DECIDING NO IN TERMS"
1300 PRINT"OF THAT FACTOR ONLY.":PRINT
1310 PRINT"   LET'S GIVE ";L$(1)
1320 PRINT"A VALUE OF 10 ON EVERY SCALE.":PRINT
1330 IF T<>3 THEN PRINT"   THEN EVERY OTHER ";T$
1340 IF T=3 THEN PRINT"   THEN DECIDING NO"
1350 PRINT"WILL BE ASSIGNED A VALUE HIGHER OR"
1360 PRINT"LOWER THAN 10.  THIS VALUE DEPENDS ON"
1370 PRINT"HOW MUCH YOU THINK IT IS BETTER OR"
1380 PRINT"WORSE THAN ";L$(1)
1390 PRINT:FOR J=1 TO NF
1400 PRINT"   ------------------"
1410 PRINT"   CONSIDERING ONLY ";F$(J);" AND"
1420 PRINT"ASSIGNING 10 TO ";L$(1);" ;"
1430 PRINT"WHAT VALUE WOULD YOU ASSIGN TO"
1440 PRINT:FOR K=2 TO NI
1450 PRINT L$(K);:INPUT"? ";C(K,J):PRINT:IF C(K,J)>=0
      THEN 1470
1460 PRINT"  --NEGATIVE VALUES NOT LEGAL--":GOTO 1450
1470 NEXT:PRINT:C(1,J)=10:NEXT
1500 FOR J=1 TO NF:Q=0:FOR K=1 TO NI
1510 Q=Q+C(K,J):NEXT:FOR K=1 TO NI
1520 C(K,J)=C(K,J)/Q:NEXT:NEXT
1530 FOR K=1 TO NI:D(K)=0:FOR J=1 TO NF
1540 D(K)=D(K)+C(K,J)*V(J):NEXT:NEXT
1550 MX=0:FOR K=1 TO NI
1560 IF D(K)>MX THEN MX=D(K)
1570 NEXT:FOR K=1 TO NI:D(K)=D(K)*100/MX:NEXT
1600 FOR K=1 TO NI:Z(K)=K:NEXT:NM=NI-1
1610 FOR K=1 TO NI:FOR J=1 TO NM:N1=Z(J):N2=Z(J+1):IF D(
     N1)>D(N2) THEN 1630
1620 Z(J+1)=N1:Z(J)=N2
1630 NEXT:NEXT:J1=Z(1):J2=Z(2):DF=D(J1)-D(J2):GOSUB
     2000
1700 PRINT L$(J1);
1710 PRINT" COMES OUT BEST"
1720 IF DF<5 THEN PRINT"BUT IT'S VERY CLOSE.":GOTO 1800
1730 IF DF<10 THEN PRINT"BUT IT'S FAIRLY CLOSE.":GOTO
     1800
```

```
1740 IF DF<20 THEN PRINT"BY A FAIR AMOUNT.":GOTO 1800
1750 PRINT"QUITE DECISIVELY."
1800 PRINT:PRINT"HERE IS THE FINAL LIST IN ORDER.":PRINT
1810 PRINT L$(J1);" HAS BEEN"
1820 PRINT"GIVEN A VALUE OF 100 AND THE OTHERS"
1830 PRINT"RATED ACCORDINGLY.":PRINT
1840 PRINT"   --------------------":PRINT
1850 FOR J=1 TO NI:Q=Z(J):PRINT D(Q);TAB(16):L$(Q):NEXT
1860 END
2000 FOR J=1 TO 500:NEXT
2010 HOME:PRINT TAB(16);"DECIDE":PRINT:RETURN
2100 FOR J=1 TO 3000:NEXT:RETURN
```

修改程式

1. 程式內以“END”字來標示輸入列表的結束。若您想以其他字取代（或許因爲“END”字與列表內的項目名稱相衝突），可更改 180 行內 E$ 的設定。例如，使用“DONE”字來取代，則改 180 行爲：

```
180 E$="DONE"
```

2. 程式內 2100 行是常見的躭擱時間程式技巧，若您感覺到步驟進行得太快，可將此行的數字 3000 放大一些。譬如：

```
2100 FOR J=1 TO 5000:NEXT:RETURN
```

3. 此程式目前最多可接受 9 個決斷選擇，與 9 個考慮因素。若您需要更多項目，可增加 160 行的陣列空間宣告。每個陣列空間值必須比程式實際所

需空間多一。因此，若使用 14 頁，則160行爲:

```
160 DIM L$(15),F$(15),V(15),C(15,15),D(15),Z(15)
```

主流程

150－180	設定變數與陣列空間
200－350	區分決斷的種類
400－490	輸入或設定T$
500－730	輸入決斷可能的選擇列表
750－950	輸入考慮因素列表
1000－1110	輸入每個考慮因素的比值
1200－1470	輸入相對於每個考慮因素的決斷可能選擇的比值
1500－1570	計算每個決斷選擇的分數
1600－1630	依照分數高低編排決斷選擇的順序
1700－1860	顯示結果
2000－2010	副程式，清除螢幕與顯示標題
2100	躭擱時間副程式。

主要變數

NI	決定選擇的個數
L$	決斷選擇的字串陣列

NF	考慮因素的個數
F$	考慮因素的字串陣列
V	考慮因素的比值陣列
A	指到關鍵因素的註標
C	相對於每個考慮因素的決斷選擇的比值陣列
T	決斷類型（1＝物品項目，2＝行爲動作，3＝是或否）
T$	決斷類型的字串名稱
E$	標示輸入列表結束的字串
J，K	迴圈指標
R$	使用者回答字串
Q，N1，N2	工作變數
D	決斷可能選擇的分數陣列
MX	所有決斷選擇最高分數
DF	兩個最佳決斷選擇之間的分數差
Z	排列好的決斷選擇陣列

研究建議

1. 允許使用者複查數值輸入，若需要的話，可修改數值輸入。
2. 欲洞察決斷的可靠度，可做敏感度的分析。其方法爲針對相同的決斷多次執行此程式。每次執行時，更改一個輸入值（通常爲您最沒把握的輸入

值）。來看結果如何改變，可依此決定那個考慮因素爲關鍵因素。若以現有的程式，此方法必須多次完全從頭執行此程式。試修改此程式，俾使使用者能夠於產生輸出結果後又能更改輸入值，然後再依新值計算新的輸出結果（注意：一旦新的輸入值給定後，會破壞不少輸入陣列內容。因之，此項修正必須將原來的輸入值抄到新陣列，以備將來的複習）。

借貸(LOAN)

計算貸款的利息及償付方式

目　的

向銀行貸款最頭痛的一點就是不易完整計算出您能選擇的貸款方式。當您向合作社辦理汽車貸款購買轎車，也許有兩種分期付款辦法：一為 36 個月，另一為 48 個月分期付款。當您貸款購買房地產時，若每期付款金額能增高的話，則能減低利息。何種方式較合算？每月付款金額的影響多大？貸款本金減少的速度影響多大？每次付款內有多少是利息，其是否免稅？

您必須知道這些問題的答案才能做最好的決定。此程式提供您所需的這些資料。

使用方法

程式首先詢問您欲借貸的金額。金額以元爲單位，超過一千萬元的貸款程式駁回（若您欲借如此鉅額，就能夠雇用一個會計師幫您計算）。然後詢問此貸款的年利率。依百分比的型式輸入此資料，譬如“10.8”。接著詢問分期付款的期次年限。若爲5年貸款輸入 60 。若爲 30 年抵押貸款，輸入 360 。程式接著顯示以上資料然後計算每月付款金額，使得在付款年限內能付淸貸款，且每月的付款金額相同。

此時您有四種選擇。第一種爲顯示月際分析。此爲以月爲單位，顯示每次付款後貸款本金的狀態。月際分析所顯示的四欄分別爲每月的貸款償付號碼（或稱月號碼），本月償付後的貸款平衡值、本次償付的利息金額、以及到目前累積的利息。螢幕一次可顯示 16 行資料，然後您可按入T鍵以取得最後貸款本金，或按入其他鍵，以顯示下面 16 個月的貸款情形。

第二種選擇爲終結月際償付。另一種很合理抵押貸款方法爲每月償付較少金額，然後在最後以大款項付淸貸款。您可使用此第二種選擇來嘗試不同的月付方式，看看其對最後的鉅款付淸有何影響。使用終結月際償付後，您須接著使用第一種選擇以獲得新的月際償付的月際分析與最後總和。

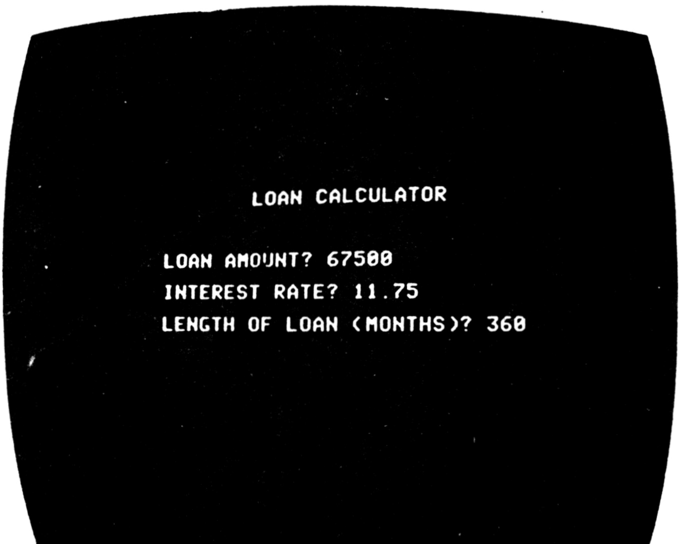
第三種選擇爲從頭開始。也許您只是比較不同的貸款方式的每月付款情形，此時就必須使用此種選擇。

第四種選擇爲終止程式執行。

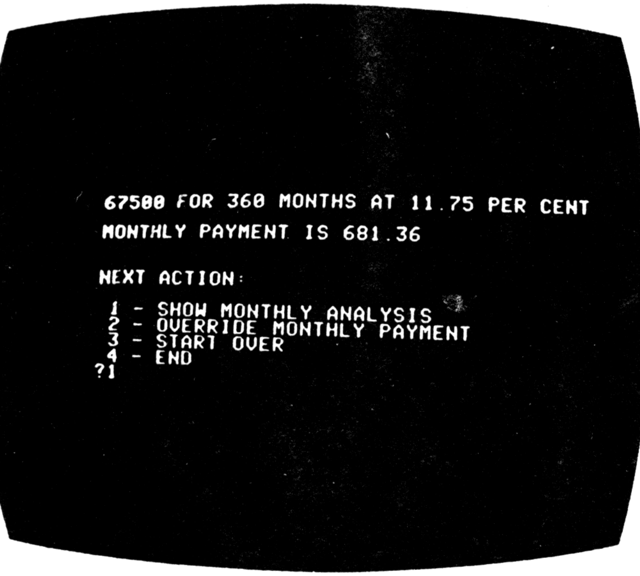
順便提及，也許您的債主所算的每月償付金額與我們此處所計算的有些誤差。我們可認定程式所做的計算較精確。

註：見書前序。

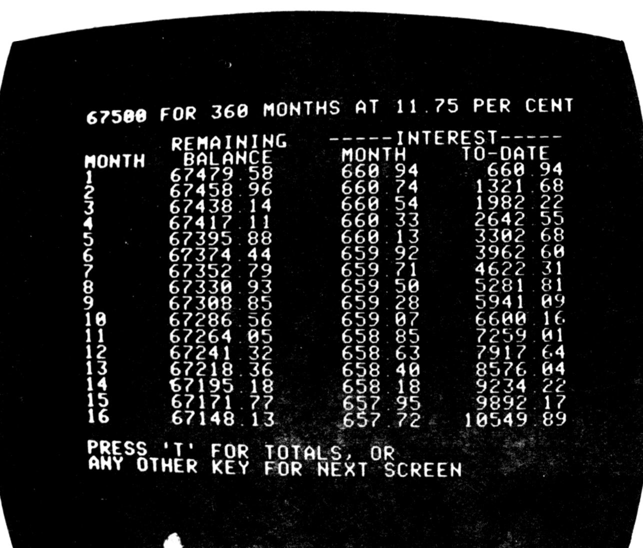
執行範例



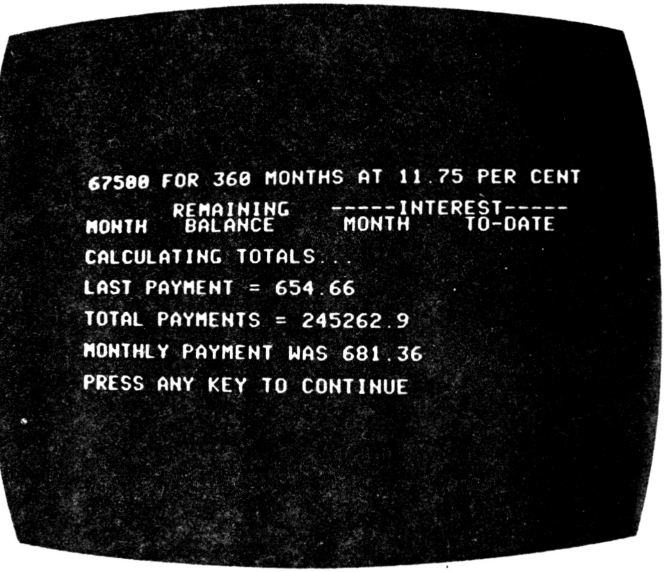
操作者輸入有關借貸的三項資料。



程式回答每月應償付的金額，然後詢問操作者下一個動作。操作者要求月際分析。



程式回答前 16 個月的貸款情形，然後等待。



操作者按“ T ”鍵，幾秒後，程式顯示貸款的總和資料。

程 式

```
100 REM: LOAN CALCULATOR
110 REM: COPYRIGHT 1980 BY TOM RUGG AND PHIL FELDMAN
120 BL$="          ":TEXT:HOME
130 PRINT TAB(8);"LOAN CALCULATOR"
140 PRINT:PRINT:PRINT
150 INPUT"LOAN AMOUNT? ";A
155 GOSUB 1000:IF A=0 THEN 150
160 PRINT:INPUT"INTEREST RATE? ";R
170 PRINT:INPUT"LENGTH OF LOAN (MONTHS)? ";N
180 R=ABS(R):N=INT(N):M=R/1200:PRINT
190 GOSUB 800
199 REM:'UP ARROW' IS SHIFTED N
200 U=(1+M)^N
```

```
210 P=A*M*W/(W-1)
220 P=INT(P*100+.99):P=P/100
230 PRINT"MONTHLY PAYMENT IS ";P
240 FP=P:PRINT:PRINT
250 PRINT"NEXT ACTION:"
260 PRINT
270 PRINT" 1 - SHOW MONTHLY ANALYSIS"
280 PRINT" 2 - OVERRIDE MONTHLY PAYMENT"
290 PRINT" 3 - START OVER"
300 PRINT" 4 - END"
310 INPUT C
320 ON C GOTO 440,400,120,370
330 PRINT"CHOICES ARE 1, 2, 3, AND 4"
340 GOTO 250
370 END
400 PRINT:INPUT"MONTHLY PAYMENT? ";P
410 GOTO 240
440 GOSUB 450:GOTO 510
450 GOSUB 800
460 PRINT TAB(8);"REMAINING";TAB(20);
470 PRINT"-----INTEREST-----"
480 PRINT"MONTH";TAB(9);"BALANCE";TAB(21);
490 PRINT"MONTH    TO-DATE"
500 RETURN
510 B=A*100:TT=0:TP=0:L=0:P=P*100:R$=""
520 FOR J=1 TO N
530 T=M*B
540 T=INT(T+.5)
550 IF J=N THEN P=B+T
560 TP=TP+P:B=B-P+T:TT=TT+T
565 IF B<0 THEN GOSUB 2000
570 IF R$="T" THEN 660
580 W=B:GOSUB 900:B$=S$
590 W=T:GOSUB 900:T$=S$
600 W=TT:GOSUB 900:TT$=S$
610 PRINT J;TAB(5);B$;T$;TT$
615 IF B=0 THEN J=N:GOTO 630
620 L=L+1:IF L<16 THEN 660
630 PRINT:PRINT"PRESS 'T' FOR TOTALS, OR"
640 PRINT"ANY OTHER KEY FOR NEXT SCREEN"
650 GET R$:IF R$="" THEN 650
655 L=0:GOSUB 450:IF R$="T" THEN PRINT:PRINT
    "CALCULATING TOTALS..."
660 NEXT
670 PRINT:PRINT"LAST PAYMENT = ";P/100
```

```
680 PRINT:PRINT"TOTAL PAYMENTS = ";TP/100
690 PRINT:PRINT"MONTHLY PAYMENT WAS ";FP
710 PRINT:PRINT"PRESS ANY KEY TO CONTINUE"
720 GET R$
730 P=FP:GOTO 240
800 HOME
810 PRINT A;" FOR ";N;" MONTHS AT ";R;" PER CENT"
820 PRINT
830 RETURN
900 W=INT(W)
905 S$=STR$(W):K=LEN(S$)
910 IF K=1 THEN S$=BL$+".0"+S$:RETURN
920 IF K=2 THEN S$=BL$+"."+S$:RETURN
930 D$="."+RIGHT$(S$,2)
950 S$=LEFT$(S$,K-2)+D$
960 S$=LEFT$(BL$,10-K)+S$
970 RETURN
1000 A=ABS(A):A=INT(A)
1010 IF A<10000000 THEN RETURN
1020 PRINT"TOO LARGE"
1030 A=0:RETURN
2000 P=P+B:TP=TP+B:B=0
2010 RETURN
```

修改程式

1. 在月際分析時，螢幕每次顯示的資料行數可經由更改程式內 620 行的常數 16 而改變。
2. 欲在月際分析的每個顯示螢幕的標題上包括每月償付金額，可插入下面一行指令：

```
815 IF FP <> 0 THEN PRINT"MONTHLY PAYMENT IS";FP
```

主流程

120－170	顯示標題，輸入貸款資料
200－230	計算及顯示每月償付
250－370	詢問下一步驟。跳到適當的副程式
400－410	輸入終結月際償付
440－730	計算及顯示月際分析
800－830	副程式，清除螢幕及顯示貸款資料於上方
900－970	副程式，將整數數值轉換成有十進位分點的等長字串
1000－1030	檢查貸款金額（大小限制及以元爲單位）
2000－2010	副程式，處理提早償付

主要變數

A	貸款金額
R	利率（百分比）
N	貸款期數年限（以月爲單位）
M	每月利率（並非百分比）
W	工作變數
P	每月償付金額
FP	首次每月償付
C	下個步驟的選擇

B	貸款所剩平衡值
TT	目前累積利息
TP	目前累積付款
L	螢幕顯示的資料行數
R$	操作員回答按入的字串
J	迴圈的工作變數
T	每月利息
B$	B的字串格式
T$	T的字串格式
TT$	TT的字串格式
S$	工作字串
D$	工作字串
K	工作變數

·研究建議

1. 顯示更豐富的借貸分析。譬如，除了全部償付總額外，更有償付總額與貸款本金之比（ TP除以 A）。
2. 修改程式使能分析一段範圍的利率或貸款期數之每月償付情形。例如，若操作員輸入利率9.5，則程式可計算顯示出利率8.5、9.0、9.5、10.0與10.5的每月償付情形。

耗油量(MILEAGE)

分析你的車輛耗油率

目 的

對於汽車、機車駕駛員而言，車輛的耗油量為相當切身的問題。此程式可幫助駕駛人統計車輛耗油量、行駛哩數與耗油率。以 DATA 敍述來儲存車輛的“資料檔”。因此若加入新資料，可直接修改儲存程式的主檔即可。程式可於汽車加滿油後，計算耗油率（MPG，哩/加侖），整理出一個行車日誌，可看出車輛耗油率、性能的變化。

使用方法

在程式開始執行前，您必須先輸入您的車輛以往耗油狀況。此份資料置於程式 1000 行起的 DATA 敍述。在每次加滿油之後，記錄日期、車輛里程表、

與購油的加侖量。

DATA敍述的格式如下：

行號 DATA 日期，里程表，加油量

注意事項爲：每個敍述的行號必須隨著日期的增加而增加。第一個行號爲 1000 ，然後下一行的行號差爲 5 或 10 ，這可方便以後於更正錯誤時插入新行。

每行 DATA 敍述含三欄，各欄必須以逗點分開。第一欄爲加油日期，是字串型式，若超過 8 個字母則只接受前 8 個字母。我們建議的日期格式爲月／日／年，即 12／25／80 爲 1980 年 12 月 25 日。日期後必須以逗點區分，然後爲里程表與加油量（以加侖爲單位）。

若您記錄車輛資料時，遺漏了某欄資料，在填入 DATA 行時，可採取下面方法：謹愼地猜測此欄的值或乾脆將此欄空下來。輸入空欄的方法如下：若爲日期欄，則 DATA 後鍵入的第一個非空白字爲逗點；若爲里程表欄，則日期後連打兩個逗點然後打加油量；若爲加油量欄，則里程表後直接打一個逗點即可。程式即能辨認這些特別的格式。總言之，每個 DATA 行必須恰好有兩個逗點。

一旦資料輸入之後，您可將此程式錄在磁帶或磁碟上。以後若有新資料進來，可搬入舊程式於主記憶體中加入新 DATA 行，然後再儲存此新程式。

輸入適當的資料後，可開始執行程式，主流程爲中心命令狀態，接受操作者的命令後，跳到三個副程

式其中一個。當副程式執行完後，控制權又回到主流程接受下一個命令。三個副程式簡單敍述如下：

驗證DATA敍述

此副程式審視 DATA 敍述各欄的資料，尋找資料的錯誤。驗證里程表的值是否太高或太低（範圍設定為 0 到 999999 之間），或加油量是否太高或太低（範圍設定為 0 到 9999 之間）。檢查里程表的值是否隨著日期的增加而增加。若發現資料錯誤，立刻顯示適當的錯誤訊息。若發現錯誤的 DATA 行（譬如，一行超過三欄，或在數值欄內發現文字字串），則錯誤訊息會指出錯誤的行號。若所有的資料都正確無誤，則副程式會顯示資料的起始日期與終止日期與資料錄個數。然後要求操作者鍵入任意鍵以回到主流程命令狀態。

顯示耗油量資料

此副程式計算耗油率（哩 / 加侖），將所有資料依列表格式顯示出來。列表一共有四欄，除了前述三欄外，多加入耗油率欄，且數值一概四捨五入。當列表資料填滿整個螢幕後，副程式會催促操作者按入任意鍵以繼續印出列表。當所有資料印完後，按入任意鍵即進入主流程的命令狀態。若 DATA 錄有嚴重錯誤，會印出錯誤訊息終止程式執行。

終止程式

此副程式終止程式執行並回到電腦 BASIC 狀態。

執行範例

```
]1000 DATA 9/28/79,51051.1,14.6
]1010 DATA 10/6/79,51299.7,13.8
]1020 DATA 10/17/79,51553.8,13.1
]1030 DATA 10/29/79,51798,13.7
]1040 DATA 11/5/79,52041.9,13.3
]RUN
                 MILEAGE

COMMANDS
 1 - VERIFY DATA STATEMENTS
 2 - DISPLAY MILEAGE INFORMATION
 3 - TERMINATE PROGRAM

 ENTER COMMAND BY NUMBER? 1

DATA FILE DISPOSITION

DATE OF FIRST DATA RECORD: 9/28/79
DATE OF LAST DATA RECORD: 11/5/79

5 DATA RECORDS FOUND

ALL DATA PROCESSED
```

```
HIT ANY KEY TO RESUME COMMAND MODE
(a key is duly pressed)

                MILEAGE

COMMANDS
 1 - VERIFY DATA STATEMENTS
 2 - DISPLAY MILEAGE INFORMATION
 3 - TERMINATE PROGRAM

 ENTER COMMAND BY NUMBER? 2

  DATE      ODOMETER    GALLONS    MPG
9/28/79     51051.1      14.6       0
10/6/79     51299.7      13.8      18
10/17/79    51553.8      13.1      19.4
10/29/79    51798        13.7      17.8
11/5/79     52041.9      13.3      18.3

ALL DATA PROCESSED
HIT ANY KEY TO RESUME COMMAND MODE

                MILEAGE

COMMANDS
 1 - VERIFY DATA STATEMENTS
 2 - DISPLAY MILEAGE INFORMATION
 3 - TERMINATE PROGRAM

 ENTER COMMAND BY NUMBER? 3

]
```

程 式

```
100 REM: MILEAGE
110 REM: COPYRIGHT 1980 BY PHIL FELDMAN AND TOM RUGG
200 TEXT:HOME
210 PRINT:PRINT TAB(16);"MILEAGE"
```

```
220 PRINT:PRINT"COMMANDS"
230 PRINT" 1 - VERIFY DATA STATEMENTS"
240 PRINT" 2 - DISPLAY MILEAGE INFORMATION"
250 PRINT" 3 - TERMINATE PROGRAM"
260 PRINT:INPUT" ENTER COMMAND BY NUMBER? ";C
270 C=INT(C):IF C<1 OR C>3 THEN 200
280 ON C GOTO 300,700,290
290 END
300 HOME:PRINT:RESTORE
310 PRINT"DATA FILE DISPOSITION":PRINT
320 N=0:DD=-.001
330 ONERR GOTO 500
340 READ A$,D,G
350 POKE 216,0:N=N+1
360 IF N=1 THEN PRINT"DATE OF FIRST DATA RECORD: ";A$
370 IF D>=0 AND D<=999999 THEN 400
380 PRINT"--BAD ODOMETER VALUE OF ";D
390 PRINT"     AT DATE: ";A$
400 IF D>DD THEN 440
410 PRINT"--INCONSISTENT ODOMETER VALUES"
420 PRINT"     ODO READS ";D;" AT DATE: ";A$
430 PRINT"     YET READS ";DD;" AT PREVIOUS DATE"
440 IF G>=0 AND G<=9999 THEN 470
450 PRINT"--BAD GALLONS VALUE OF ";G
460 PRINT"     DETECTED AT DATE: ";A$
470 DD=D:GOTO 330
500 LN=PEEK(218)+256*PEEK(219)
510 EC=PEEK(222):POKE 216,0
520 IF EC=16 THEN 560
530 IF EC=42 THEN 590
540 PRINT:PRINT"--ERROR # ";EC;" HAS BEEN DETECTED"
550 PRINT"     AT LINE ";LN:END
560 PRINT"--BAD DATA RECORD DETECTED"
570 PRINT"     AT OR BEFORE LINE ";LN
580 PRINT"--PROGRAM ABORTED":END
590 IF N>0 THEN 620
600 PRINT:PRINT"--NO DATA FOUND"
610 PRINT"--PROGRAM ABORTED":END
620 IF C=2 THEN 650
630 PRINT"DATE OF LAST DATA RECORD: ";A$
640 PRINT:PRINT N;" DATA RECORDS FOUND"
650 PRINT:PRINT"ALL DATA PROCESSED"
660 PRINT"HIT ANY KEY TO RESUME COMMAND MODE"
670 GET T$:GOTO 200
```

```
700 GOSUB 820:RESTORE:DD=-1:N=0
710 ONERR GOTO 500
720 READ A$,D,G:POKE 216,0:N=N+1:A$=LEFT$(A$,8)
730 R=D:GOSUB 860:D=R:LD=18-L:IF INT(D)=D THEN LD=LD-2
740 R=G:GOSUB 910:G=R:LG=27-L:IF INT(G)=G THEN LG=LG-2
750 IF DD<0 OR G=0 THEN M=0:GOTO 770
760 M=(D-DD)/G
770 R=M:GOSUB 960:M=R:LM=38-L:IF INT(M)=M THEN LM=LM-2
780 DD=D:GOSUB 790:GOTO 710
790 PRINT TAB(1);A$;TAB(LD);D;TAB(LG);G;TAB(LM);M
800 K=K+1:IF K<20 THEN RETURN
810 PRINT:PRINT"HIT ANY KEY TO CONTINUE";:GET T$:PRINT
820 HOME:PRINT
830 PRINT TAB(1);"  DATE    ODOMETER   GALLONS     MPG"
840 K=0:RETURN
860 IF R>=0 AND R<=999999 THEN 890
870 PRINT:PRINT"ERROR IN ODOMETER DATA AT DATE:";A$
880 END
890 R=R*10+.5:R=INT(R)/10
900 L=LEN(STR$(R)):RETURN
910 IF R>=0 AND R<=9999 THEN 940
920 PRINT:PRINT"ERROR IN GALLONS DATA AT DATE: ";A$
930 END
940 R=R*10+.5:R=INT(R)/10
950 L=LEN(STR$(R)):RETURN
960 IF R<0 OR R>9999 THEN R=0
970 R=R*10+.5:R=INT(R)/10
980 L=LEN(STR$(R)):RETURN
```

修改程式

1. 若您想替資料檔定一個名字，然後在命令狀態中印出，可更改 210 行，且加入 205 行如下：

```
205 B$="VOLVO 1979"
210 PRINT:PRINT"MILEAGE FOR:";B$
```

只要把檔案名稱存入 205 行中的 B$ 內即可。

2. 程式內使用了Apple電腦特殊的ONERR GOTO 敍述，此可偵查一些預期的錯誤。若有一個不預期的錯誤發生，程式可能於印出錯誤訊息後執行失敗，譬如：

-- ERROR # 255 HAS BEEN DETECTED AT LINE 340

這些錯誤號碼及其意義可見您的Applesoft手册在 ONERR GOTO 處的說明。若您發生錯誤而不曉得其原因，可做以下的修改，然後再重新執行程式，即能得到正常的 BASIC 錯誤訊息

```
330 REM
710 REM
```

主流程

200 − 280	命令狀態。顯示可用的副程式，然後依操作者的選擇，跳到適當的副程式。
290	副程式，終止程式執行。
300 − 470	副程式，驗證 DATA 敍述。
500 − 670	處理讀入 DATA 敍述時所發生的錯誤
700 − 780	副程式，顯示耗油量資料
790 − 840	副程式，印出結果
860 − 900	副程式，里程表值四捨五入

910 − 950	副程式，加油量四捨五入
960 − 980	副程式，耗油率四捨五入
1000	使用者建立的 DATA 敍述

主要變數

C	命令標示（1＝驗證 DATA ，2＝顯示耗油率，3＝終止執行）
N	讀入的資料錄個數
A$	目前資料錄的日期
DD	前一資料錄的里程表值
D	目前里程表值
G	目前耗油率
LD	印里程表值的位置
LG	印加油量的位置
LM	印耗油率的位置
T$	工作字串變數
L	字串長度
R	四捨五入前的數
K	由 HOME 算起，印的行數
EC	錯誤碼
LN	偵出的錯誤行號

研究建議

1. 計算及印出整個資料檔的平均耗油率。此即行駛的總里程值除以總耗油量。總里程值為第一個 DATA 敍述的里程表值與最後里程表值之差。總耗油量則為由第二個 DATA 敍述的加油量到最後 DATA 敍述的加油量之總和。
2. 添加一個命令選擇，做資料檔的特定子集合的統計分析。操作者可輸入一個起始日期與終止日期，然後副程式可計算出此範圍的平均耗油率、總里程值、總耗油量……等等。
3. 設計一個圖形副程式，以圖形顯示耗油率。
4. 在每個 DATA 敍述上加上一個新欄——每次加油的費用。然後計算出汽油總價與哩/元等等。

問卷統計 (QUEST / EXAM)

幫你做問卷評閱及分析

目　的

若您曾因研究或學習的需要而做過問卷調查，應已深深體會到問卷統計是個煩瑣耗時的工作。尤其當您想收集分析出每個問題各個答案的人數分佈情形時，更會感到耗時煩惱。有此程式的幫助，您只要提供資料，電腦即會幫您做此工作。

使用方法

首先，輸入問卷的題目數目（程式限制最大不能超過 35 題）。然後輸入每個問題的選擇個數，其爲 2 到 9 的整數。問卷的答案可能爲介於 1 到此數間的整數，或者空白。

然後輸入問卷的最大張數。必須確使您眞正處理

的問卷個數不大於此最大張數。

最後，詢問您是否欲對每一份問卷設定名字。若此問卷為一種測驗，此可方便於了解各學生的答題情形。

接著，程式詢問您問卷的標準答案。若您的問卷是考試，要計算成績，就必須提供標準答案。程式會顯示引導數字" guide numbers "來幫助您輸入正確答案。若為調查問卷，即無標準答案，可直接按 RETURN 鍵。

現在，程式開始要求您輸入每份問卷的答案。同樣地，輸入資料的上頭會顯示引導數字，方便您針對題號輸入問卷的答案。若問卷上某題沒寫答案，則留一個空格。然而，若第一題或最後一題為空白，則必須將您輸入的答案以"引號"括起來。雖然如此一來會與上方參考的引導數字不符合，但您會習慣的。見以下另一種表示方式。

若您輸入資料時發生錯誤，程式會告訴您，叫您重新輸入。通常易犯的錯誤為輸入不正確的問卷答案或按入錯誤的鍵。應注意每個答案為空白或 1 到此題選擇個數之間。

順便提及，您也可避免對沒答的題目輸入空格。假設問卷上的問題至多有 5 個選擇，則一旦碰到沒答的問題，可輸入 6 以取代空格。

若您表示欲對每一問卷設定一個名字。程式會在您輸入一問卷答案後詢問此問卷之名字。在名字字串中不要使用逗點（除非整個名字已用引號括好）。

假如您提供了標準答案，則程式於接受一張問卷後即顯示正確的題數與百分比，接著才詢問下一張問卷。當您已輸入所有問卷答案，則按入 RETURN 鍵以取代答案字串。

此時，程式接著顯示五個選擇，要您決定下一個步驟。以下為扼要的說明，您可以親身試驗其操作情形。

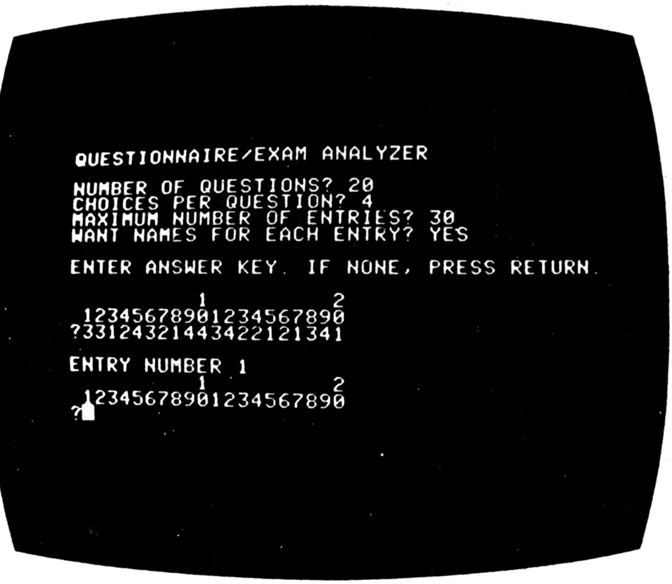
第一個選擇讓您分析每個問題，看看各個選擇答案的人數，各個答案的分配百分比。若為考試測驗，則正確答案以“ RIGHT ”字表示。

第二個選擇讓您回頭提供更多的問卷。此功能可讓您先輸入部分問卷，做這些部分問卷的分析工作後，再繼續輸入其他問卷。

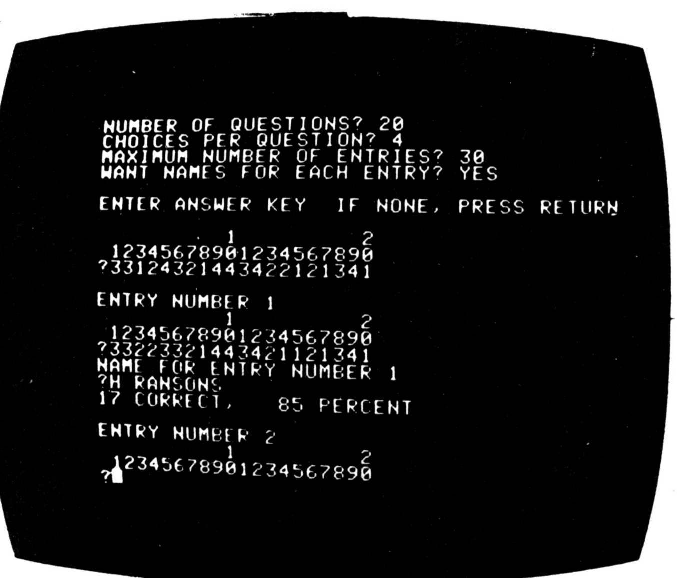
第三個選擇讓您複習先前輸入的問卷答案以及標準答案，此可讓您檢查輸入的問卷是否有重覆、忽略或錯誤之處。

第四個選擇為重新執行程式。第五個選擇為終止程式執行。

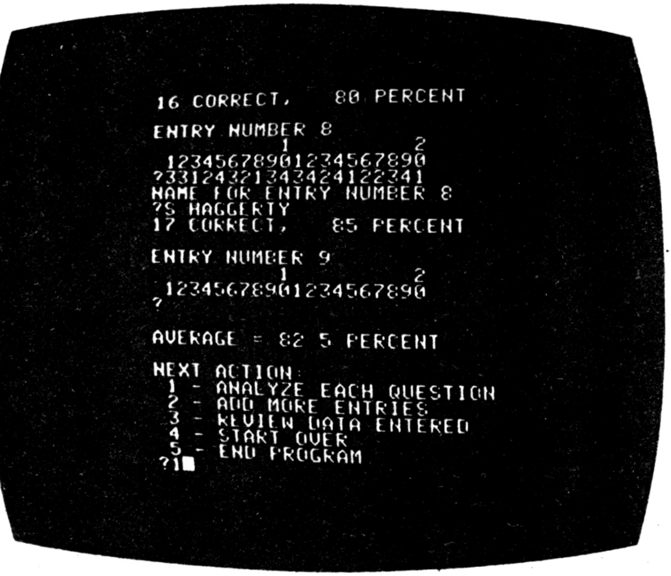
執行範例



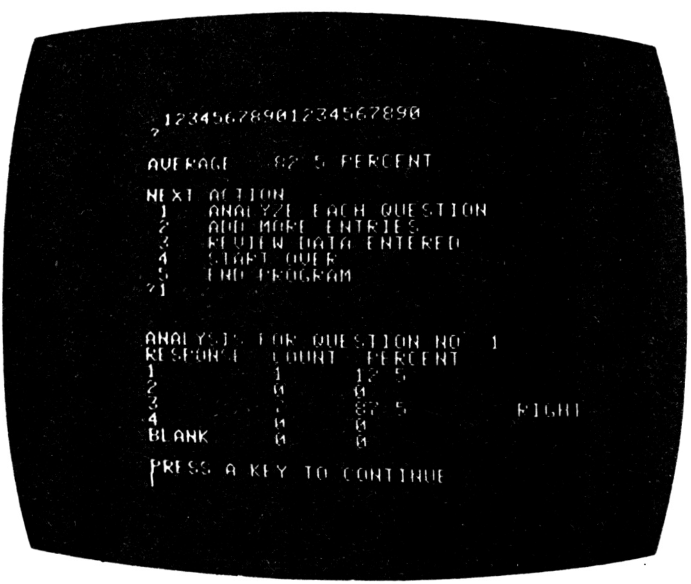
操作者提供問卷的資料，程式等待第一張問卷的答案



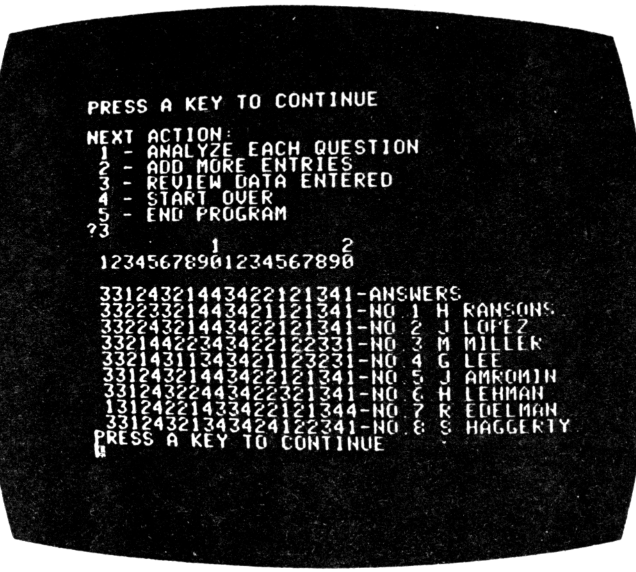
輸入第一張問卷的答案，接著爲學生姓名，程式回答答對的題數及百分比。



不久後，操作者不輸入第九個學生的問卷，而直接打 RETURN 鍵，表示沒有問卷了。程式顯示平均答對率。然後顯示下一動作的選擇菜單，操作者選1。



程式提供問卷第一個問題的分析，然後等待操作者按鍵。注意，有7個學生答正確答案3。



不久後，操作者要求選擇 3。其印出輸入的學生各問卷答案

程 式

```
100 REM: QUESTIONNAIRE/EXAM
110 REM: COPYRIGHT 1980 BY TOM RUGG AND PHIL FELDMAN
120 CLEAR:TEXT:HOME
130 PRINT"QUESTIONNAIRE/EXAM ANALYZER":PRINT
140 E$="** ERROR. RE-ENTER. **":P$=
    "PRESS A KEY TO CONTINUE"
150 INPUT"NUMBER OF QUESTIONS? ";Q
160 Q=INT(Q):IF Q<1 OR Q>35 THEN PRINT E$:GOTO 150
170 INPUT"CHOICES PER QUESTION? ";C
180 C=INT(C):IF C<2 OR C>9 THEN PRINT E$:GOTO 170
185 DIM C(C)
190 INPUT"MAXIMUM NUMBER OF ENTRIES? ";N
200 N=INT(N):IF N<1 THEN PRINT E$:GOTO 190
205 DIM Q$(N+1)
210 INPUT"WANT NAMES FOR EACH ENTRY? ";R$
```

```
220 R$=LEFT$(R$,1):IF R$="N" THEN 250
230 IF R$<>"Y" THEN PRINT E$:GOTO 210
240 DIM N$(N)
250 PRINT:PRINT
    "ENTER ANSWER KEY. IF NONE, PRESS RETURN."
260 GOSUB 900:C$=STR$(C):PRINT TAB(1);
310 INPUT A$:IF LEN(A$)=0 THEN 340
320 IF LEN(A$)<>Q THEN PRINT E$:GOTO 250
330 T$=A$:GOSUB 850:IF T$="B" THEN PRINT E$:GOTO 250
340 K=1
350 R=0:PRINT:PRINT"ENTRY NUMBER ";K
360 GOSUB 900:PRINT TAB(1);
370 INPUT Q$(K):W=LEN(Q$(K))
380 IF W=0 THEN 500
390 IF W<>Q THEN PRINT E$:GOTO 350
400 T$=Q$(K):GOSUB 850:IF T$="B" THEN PRINT E$:GOTO 350
410 IF R$="N" THEN 430
415 PRINT"NAME FOR ENTRY NUMBER ";K
420 INPUT N$(K)
430 IF LEN(A$)=0 THEN 480
440 FOR J=1 TO Q
450 IF MID$(A$,J,1)=MID$(Q$(K),J,1) THEN R=R+1
460 NEXT
470 TR=TR+R:PRINT R;" CORRECT,";TAB(15);R*100/Q;
    " PERCENT"
480 K=K+1:IF K<=N THEN 350
500 K=K-1:IF LEN(A$)=0 THEN 520
505 IF K=0 THEN 960
510 PRINT:PRINT"AVERAGE = ";TR*100/(Q*K);" PERCENT"
520 GOTO 960
530 PRINT:FOR J=1 TO Q
540 R=0:PRINT:PRINT"ANALYSIS FOR QUESTION NO. ";J
545 PRINT"RESPONSE  COUNT  PERCENT"
550 FOR L=0 TO C:C(L)=0:NEXT
560 FOR L=1 TO K:T$=MID$(Q$(L),J,1)
570 W=VAL(T$)
580 C(W)=C(W)+1
600 NEXT L
610 IF K=0 THEN 960
620 FOR L=1 TO C:PRINT L;TAB(11);C(L);TAB(17);C(L)*100/
    K;
630 IF LEN(A$)=0 THEN PRINT:GOTO 660
640 T$=STR$(L)
```

```
650 IF T$=MID$(A$,J,1) THEN PRINT TAB(29);"RIGHT":GOTO
    660
655 PRINT
660 NEXT:PRINT"BLANK";TAB(11);C(0);TAB(17);C(0)*100/K
670 PRINT:PRINT P$
680 GET T$
690 NEXT J:GOTO 960
700 L=0:GOSUB 900:PRINT:IF LEN(A$)=0 THEN 720
710 PRINT TAB(2);A$;"-ANSWERS"
720 FOR J=1 TO K
730 PRINT TAB(2);Q$(J);"-NO.";J;
740 IF R$="N" THEN PRINT:GOTO 750
745 PRINT" ";N$(J)
750 L=L+1:IF L<10 THEN 780
760 L=0:PRINT P$
770 GET T$
780 NEXT:PRINT P$
790 GET T$:GOTO 960
850 FOR J=1 TO LEN(T$):IF MID$(T$,J,1)=" " THEN 870
860 IF MID$(T$,J,1)<"1" OR MID$(T$,J,1)>C$ THEN 880
870 NEXT:RETURN
880 T$="B":RETURN
900 W=Q/10:W=INT(W):IF W<1 THEN 920
910 FOR J=1 TO W:X=J*10+1:PRINT TAB(X);J;:NEXT:PRINT
920 PRINT TAB(2);
930 FOR J=1 TO Q:PRINT RIGHT$(STR$(J),1);
940 NEXT:PRINT:RETURN
960 PRINT:PRINT"NEXT ACTION:"
970 PRINT" 1 - ANALYZE EACH QUESTION"
980 PRINT" 2 - ADD MORE ENTRIES"
990 PRINT" 3 - REVIEW DATA ENTERED"
1000 PRINT" 4 - START OVER"
1010 PRINT" 5 - END PROGRAM"
1050 INPUT T$:IF T$<"1" OR T$>"5" THEN 1070
1060 ON VAL(T$) GOTO 530,480,700,120,1100
1070 PRINT E$:GOTO 960
1100 END
```

修改程式

1. 若此程式在Apple 個人電腦有 4000 byte 可用主記憶體上執行（見附錄 1），則輸入的資料必須受一些限制。此限制大約爲每份問卷 25 個問題，每題有 5 個選擇，可容納 30 份問卷，且各問卷沒名字。假如您有 10000 byte 可用空間，即有足夠空間給 35 個問題，9 個選擇，100 份問卷，且各問卷有名字。

主流程

120 － 140	設定變數
150 － 240	處理開場白對話。決定選擇動作。安排陣列
250 － 320	取得操作員按入的回答鍵
330	檢查回答鍵的正確性
350 － 400	輸入第K份問卷答案
410 － 420	輸入第K份問卷的名字（若存在的話）
430 － 470	計算第K份問卷的答對分數（若爲考試測驗）
530 － 690	分析每個問題各個選擇答案的反應情形
700 － 790	顯示輸入的資料
850 － 880	副程式，檢查輸入資料的正確性

900 － 940	副程式，顯示幫助問卷答案輸入的引導數字
960－1100	顯示下一步驟的選擇動作，取得操作者的回答，跳到適當的副程式

主要變數

E$	錯誤訊息
P$	指示操作者按控制鍵的訊息
Q	問題題數（ 1 － 35 ）
C	每一題的選擇答案數（ 2 － 9 ）
C	對問題回答同一答案的人數陣列
N	問卷最大份數
Q$	問卷答案字串陣列
R$	若問卷無名字則設N ；否則設Y
N$	Q個名字的陣列（若R$爲Y ）
A$	標準答案字串（若非考試，則爲空白）
C$	最高合法答案的字串
K	問卷份數的計數器
R	答對的題數（若爲考試）
W	工作變數
J , L , M	迴圈變數
TR	所有問卷的答對題數
T$	工作字串變數

研究建議

1. 加上一個選擇步驟：在各試卷的答案都輸入後，可改變答案。此可在複習輸入資料後，更正錯誤的資料。
2. 加上一個選擇步驟：允許操作者於輸入所有資料後，重算一份試卷的分數，以防止輸入時之疏忽。
3. 結合書後的統計程式的部分能力於此程式內。

第二部分 教育程式：

幫助你學習

介 紹

教育方面也爲電腦科技發展衝擊得極厲害的一個領域。雖然目前電腦尚無法完全取代教師，但機器本身也有某些教師比不上的優點。它隨時可供您學習，完全配合您自己的學習速度，可輕易地處理條文背誦式知識，而且完全沒有任何個人情緒問題。

只要賦予一個優良的軟體圖書館，Apple 個人電腦就可成爲學校或家庭的優秀教師。以下爲 6 個簡單的教育程式，供您啓蒙學習。

數學（算術）可謂爲電腦的本行。NUMBERS 數字程式專爲入學前的小朋友所設計的。爲了讓小朋友們對電腦產生好感，它以遊戲的方式來教小朋友們認識數字與基本計數。ARITHMETIC 算術程式適合較懂事、已入學的小學生。它提供各種算術問題讓小朋友練習，小朋友可自行調整困難程度，使此程式

可適用好幾年。

Apple個人電腦並非只限制在數學的訓練。我們更提供了兩個程式以增進您的字彙能力。VOCAB字彙程式可幫您擴充自己的字彙。TACHIST速讀機程式可使Apple 電腦成爲一個語音學習機，幫您修正您的語音速度。

您是否曾爲通用的公制度量系統而苦惱？METRIC 公制程式即能幫您解決困擾。

是否深爲記憶力不好而苦惱？FLASHCARD記憶卡程式，可允許您自己安排記憶卡，讓您反覆練習記憶。

算術(ARITHMETIC)

替小學生出算術練習題

目　的

ARITHMETIC 算術程式爲小學生提供各種數學練習題。學生可向程式要求加法、減法、或乘法問題，而且還可指定問題的難易度爲簡單、中等或困難。此程式可適合於很大段年齡範圍的小孩子。小朋友們可於熟練一階段的練習後，進到較深的問題。程式所提供的練習題難易度分佈和類型可適於小學一年級到四年級的學生。

程式所提供的練習題是在難易度的限定下隨機形成的。此可讓學生於每次練習時，都可保持新鮮感。小朋友輸入答案後，程式就自動告訴他算對還是算錯。接著顯示正確的答案。

使用方法

首先，程式詢問學生欲做那一類的練習題，以1代表加法，2代表減法，3代表乘法。接著，程式詢問題目的難易度，分別以1、2、3來代表簡單、中等和困難。

現在，程式先清除螢幕，然後顯示所要求類型的五個練習題。學生即可開始輸入每個練習題的答案。

螢幕上以問號來催促學生輸入答案的每一位數，一次一個位數，做法就如解一般的算術問題，由左至右算過去。

開始解一練習題時，問號會顯示於答案位置的個位數上。當按入0－9的鍵後，按入的鍵值會取代問號，而問號移向此位置的左邊（即十位數）。

在整個答案完全輸入以前，鍵入的字都是以由右向左的方式輸入。鍵入一個答案後，必須按RETURN鍵，此可結束目前練習題的答案輸入，將問號移到下一個練習題的答案位置。

若直接按 RETURN 鍵來回答練習題，則程式假定您的答案爲0。此程式產生的練習題答案一概不超過三位數，若您按入四位數的答案，程式照樣接受，但立刻跳到下一個練習題。練習題的答案都爲正數。

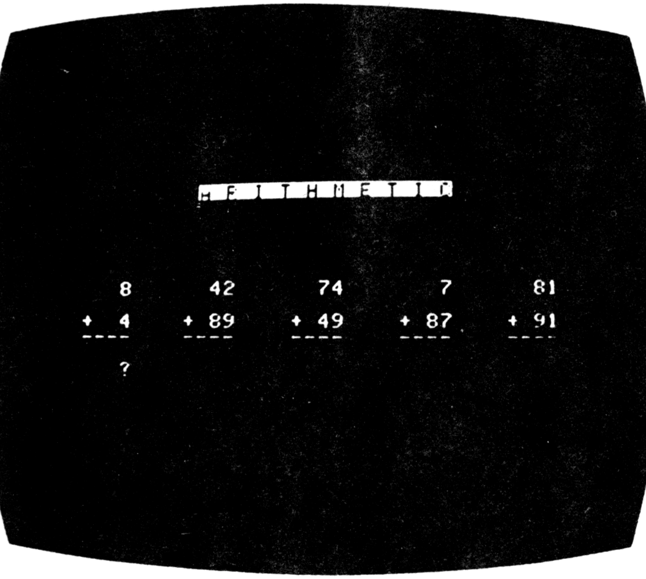
一旦學生輸入此五題的答案後，程式即顯示此五題的標準答案。同時在每題的下方顯示“ RIGHT!”或“ WRONG！”字。

然後繼續出題目，直到做完 20 個練習題。在結束前，程式計算出學生的成績。成績以答對的題數或答對題目的百分比表示。

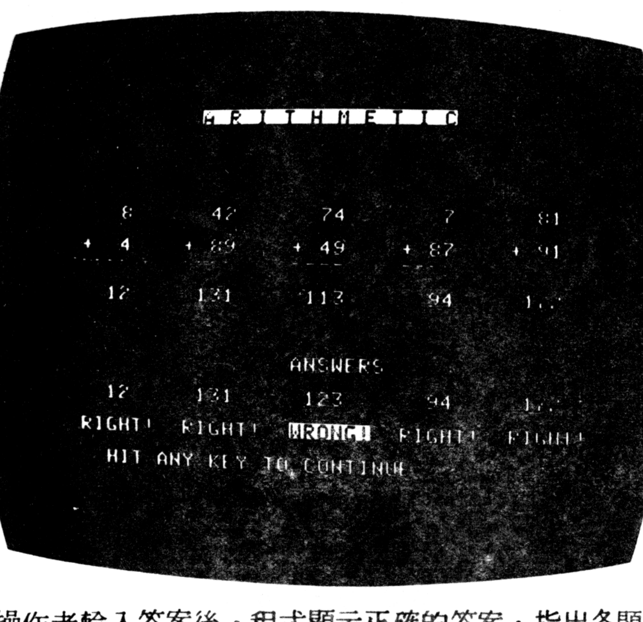
執行範例



操作者選擇困難的問題



首先顯示 5 個問題，以問號催促操作者回答第一個問題



操作者輸入答案後，程式顯示正確的答案，指出各題是否答對。然後等待操作者按任何一鍵以繼續下 5 個問題

程 式

```
100 REM: ARITHMETIC
110 REM: COPYRIGHT 1980 BY PHIL FELDMAN AND TOM RUGG
130 TEXT:K=PEEK(78)+256*PEEK(79):K=RND(-K)
150 ND=0
160 DIM A(5),B(5),C(5),G(5)
170 NP=20
180 GOSUB 910
200 PRINT:PRINT:PRINT"WHAT TYPE OF PROBLEM SHALL WE DO?"
    :PRINT
210 PRINT TAB(5);"1 - ADDITION"
220 PRINT TAB(5);"2 - SUBTRACTION"
230 PRINT TAB(5);"3 - MULTIPLICATION"
240 PRINT:INPUT"WHICH TYPE (1, 2, OR 3) ? ";R$
250 T=INT(VAL(R$)):IF T<1 OR T>3 THEN 240
260 PRINT:FOR J=1 TO 39:PRINT"=";:NEXT
270 PRINT:PRINT:PRINT"WHAT KIND SHALL WE DO"
280 PRINT:PRINT TAB(5);"1 - EASY PROBLEMS"
290 PRINT TAB(5);"2 - MEDIUM PROBLEMS"
300 PRINT TAB(5);"3 - HARD PROBLEMS"
310 PRINT:INPUT"WHAT KIND (1, 2, OR 3) ? ";R$
320 D=INT(VAL(R$))
330 IF D<1 OR D>3 THEN 310
350 ON D GOTO 360,370,390
360 GOSUB 940:GOSUB 920:GOSUB 930:GOTO 400
370 GOSUB 940:GOSUB 930:IF T=3 THEN GOSUB 960:GOSUB 920:
    GOTO 400
380 IF T<>3 THEN GOSUB 950:GOSUB 920:GOTO 400
390 GOSUB 950:GOSUB 920:GOSUB 930:IF T=3 THEN GOSUB 940:
    GOSUB 930
400 IF T<>2 THEN 450
410 FOR J=1 TO 5
420 IF B(J)>C(J) THEN R=C(J):C(J)=B(J):B(J)=R
430 NEXT
450 GOSUB 1000:GOSUB 910
600 Y=12:FOR J=1 TO 5:X=-4+J*8:GOSUB 1100:NEXT
610 FOR K=1 TO 5:X=-4+K*8:GOSUB 800:G(K)=N:NEXT
620 X=17:Y=16:GOSUB 1200:PRINT"ANSWERS"
630 Y=18:FOR J=1 TO 5:X=-4+J*8:GOSUB 1400:NEXT
640 Y=20:FOR J=1 TO 5:X=-7+J*8:GOSUB 1200
```

```
650 IF A(J)<>G(J) THEN INVERSE:PRINT"WRONG!":NORMAL:
    GOTO 670
660 PRINT"RIGHT!":NR=NR+1
670 NEXT
680 PRINT:FLASH:PRINT"  HIT ANY KEY TO CONTINUE ";
690 GET R$:NORMAL
700 ND=ND+5
710 IF ND<NP THEN GOSUB 910:GOTO 350
720 GOSUB 1500
730 END
800 N=0:M=1
810 FLASH:P$="?":GOSUB 900
820 GET R$
830 A=ASC(R$):IF A=13 AND M=1 THEN P$="0":NORMAL:GOSUB
    900:RETURN
840 IF A=13 THEN P$=CHR$(32):NORMAL:GOSUB 900:RETURN
850 IF A<48 OR A>57 THEN 820
860 V=A-48:P$=CHR$(A):NORMAL:GOSUB 900:N=N+M*V:M=M*10
870 IF M>1000 THEN RETURN
880 X=X-1:GOTO 810
900 HTAB X:VTAB Y:PRINT P$;:HTAB X:VTAB Y:RETURN
910 HOME:PRINT TAB(10);:INVERSE:PRINT
    "A R I T H M E T I C":NORMAL:RETURN
920 FOR K=1 TO 5:C(K)=INT(RND(1)*(H-L+1))+L:NEXT:RETURN
930 FOR K=1 TO 5:B(K)=INT(RND(1)*(H-L+1))+L:NEXT:RETURN
940 H=9:L=0:RETURN
950 H=99:L=0:RETURN
960 H=25:L=1:RETURN
1000 ON T GOTO 1010,1020,1030
1010 FOR J=1 TO 5:A(J)=B(J)+C(J):NEXT:RETURN
1020 FOR J=1 TO 5:A(J)=C(J)-B(J):NEXT:RETURN
1030 FOR J=1 TO 5:A(J)=C(J)*B(J):NEXT:RETURN
1100 GOSUB 1200:CU=5:CL=1:GOSUB 1300
1110 IF C(J)<10 THEN PRINT CHR$(32);
1120 PRINT C(J):GOSUB 1200:CU=3:CL=3:GOSUB 1300:IF T=1
     THEN PRINT CHR$(43);
1130 IF T=2 THEN PRINT CHR$(45);
1140 IF T=3 THEN PRINT CHR$(88);
1150 PRINT CHR$(32);:IF B(J)<10 THEN PRINT CHR$(32);
1160 PRINT B(J):GOSUB 1200:CU=2:CL=3:GOSUB 1300:FOR K=1
     TO 4:PRINT CHR$(45);
1170 NEXT:RETURN
1200 HTAB X:VTAB Y
1210 RETURN
```

```
1300 HTAB (PEEK(36)-CL+1)
1310 VTAB (PEEK(37)-CU+1):RETURN
1400 GOSUB 1200:CL=0:IF A(J)>9 THEN CL=1
1410 IF A(J)>99 THEN CL=2
1420 IF A(J)>999 THEN CL=3
1430 HTAB (PEEK(36)-CL+1)
1440 PRINT A(J):RETURN
1500 GOSUB 910:PRINT:PRINT
1510 PRINT"YOU GOT ";NR;" RIGHT"
1520 PRINT"OUT OF ";NP;" PROBLEMS
1530 P=NR/NP*100
1540 PRINT:PRINT"THAT'S ";P;" PERCENT CORRECT":RETURN
```

修改程式

1. 程式目前在每次測驗中，有 20 個練習題。您可更改題數，只要改變程式 170 行的 NP 變數即可。譬如：

```
170 NP=10
```

會使每次測驗只做 10 個題目。NP 的值必須爲 5 的倍數。

2. 在目前程式中，允許運算元素爲 0，若您不希望有此狀況產生，可更改 940 行與 950 行如下：

```
940 H=9:L=1:RETURN
950 H=99:L=1:RETURN
```

主流程

130 – 180	設定變數，顯示標題
200 – 330	詢問操作者練習題的類型
350 – 450	設定A、B、C陣列，清除螢幕
600 – 730	主程序－－顯示練習題，取得操作者的答案；顯示正確答案及評分。
800 – 880	副程式，輸入及顯示操作者的答案
900	印字副程式
910	副程式，清除螢幕及顯示標題
920 – 930	副程式，設定B、C陣列
940 – 960	副程式，設定L、H
1000–1030	副程式，由B、C陣列計算A陣列
1100–1170	副程式，顯示練習題
1200–1210	副程式，移動指標到螢幕X、Y位置
1300–1310	副程式，移動指標到由上算下來第CU行，由左算來第CL格
1400–1440	副程式，顯示正確答案
1500–1540	副程式，顯示操作者的成績

主要變數

NP	一次測驗的總題數
ND	做過的題數

NR	答對的題數
C,B,A	每個練習題的第一運算元素、第二運算元素、及正確答案
N	目前練習題的操作者答案
G	操作者答案陣列
T	要求的問題類型（1＝加法，2＝減法，3＝乘法）
D	問題的難易度（1＝簡單，2＝中等，3＝困難）
H，L	練習題中的第一個與第二個運算元素整數
M	目前處理的答案欄
R$	操作者輸入的字元
V	R$的值
A	R$的ASCII碼值
X，Y	螢幕指標的水平與垂直位置
J，K	迴圈變數與工作變數
CU,CL	向上與向左移動螢幕指標的位置數
P	答對題目的百分比
P$	欲印的字元

研究建議

1. 記下答錯的練習題然後反覆的練習。
2. 目前程式不允許負數出現於練習題中。請改寫練習題產生程序使其允許負數的出現。

3. 目前答案限制在三位數以內。然而，若可允許練習題的運算元素擴大範圍，則程式即可允許四位數的答案。請自行探究程式 350 － 450 行與940 － 960 行的處理過程，然後修改它們以允許四位數的答案。

4. 目前程式無法讓操作者修改他按入答案時的按鍵錯誤。擴充此程式，使具有此種功能。

5. 修改程式以產生除法的練習題。

記憶卡(FLASHCARD)

建立記憶卡以資背誦

目　的

人類的學習過程中，有很多事情都必須靠反覆的複習訓練方能奏效，極少人能夠過目成誦的。譬如，小時候背九九乘法表，還不是老師逼著您天天背誦，才能夠朗朗上口，謹記在心。學習外國語言，英語、日語的字母、單字，或背誦地理、歷史的地名人名事跡等，都是需要反覆背誦，方能謹記在心。

背誦這些條文式的知識有一方法就是使用記憶卡。可將資料切分爲兩部份，一份爲問卡；一份爲答卡。建立好這些卡片後，您可以反覆練習背誦問卡及答卡的內容及其相互的關係，使自己每當看到問卡後，可立即說出答卡內容。

您可不必浪費紙張製作這些記憶卡，可直接以個電腦取代。此程式可讓您建立記憶卡，反覆練習，且將其儲存於磁帶以備以後複習。

使用方法

程式目前寫好的部份是教您背誦西班牙語的字彙。先介紹如何使用此程式來練習背誦西班牙字彙，然後於“程式修改”部分教您如何建立自己的記憶卡。

程式先顯示一張記憶卡的一邊（視爲問卡），那張卡片和那一邊都是隨機選擇。您的責任就是回答此卡另一邊的內容，視爲答卡。若您輸入正確的答案，程式即回答“RIGHT ！”。若否，程式會告訴您正確的答案。然後程式會再隨機地選出另一張記憶卡與一邊，來問您下一個問題。此程式反覆提出問題，直到您按入 RETURN 鍵，通知程式不想繼續練習。程式即顯示出您一共答對了幾題，答對的百分比多少，然後給您建議應繼續練習或終止程式執行。

在練習之時，程式不會重覆出現在最近出現的四張卡片內之記憶卡。

執行範例

```
FLASHCARD PROGRAM

30 FLASHCARDS

TELL ME WHAT'S ON THE OTHER
SIDE OF EACH CARD AS I SHOW IT.

EL SUELO
```

```
?THE FLOOR

RIGHT!

AZUL

?YELLOW

NO, THE CORRECT RESPONSE IS
BLUE

ENOUGH

?BASTANTE

RIGHT!

TO EAT

? (RETURN key pressed)

2 RIGHT OUT OF 3

66.6666667 PER CENT

NEXT ACTION:
 1 - DRILL MORE
 2 - END PROGRAM

?2

]
```

程 式

```
100 REM: FLASHCARD
110 REM: COPYRIGHT 1980 BY TOM RUGG AND PHIL FELDMAN
120 TEXT:HOME:GOSUB 580
130 L=50:M=5
```

```
140 DIM F$(L),B$(L),P(M-1)
150 PRINT"FLASHCARD PROGRAM"
160 PRINT:PRINT
170 K=1:C=0:W=0
180 IF K>L THEN 240
190 READ R$:IF R$="XXX" THEN 250
200 F$(K)=R$
210 READ R$:B$(K)=R$
220 K=K+1
230 GOTO 180
240 PRINT"FLASHCARD ARRAY FULL.":PRINT
250 K=K-1
260 IF K<M THEN PRINT"NOT ENOUGH FLASHCARDS":END
270 PRINT K;" FLASHCARDS"
280 PRINT:PRINT
300 PRINT"TELL ME WHAT'S ON THE OTHER"
310 PRINT"SIDE OF EACH CARD AS I SHOW IT."
320 PRINT:PRINT
340 R=INT(K*RND(1))+1
350 FOR J=0 TO M-2
360 IF P(J)=R THEN 340
370 NEXT
390 J=RND(1):IF J>.5 THEN 420
400 PRINT F$(R):C$=B$(R)
410 GOTO 430
420 PRINT B$(R):C$=F$(R)
430 PRINT:INPUT R$
440 IF LEN(R$)=0 THEN 600
450 PRINT
460 IF R$=C$ THEN 500
470 PRINT"NO, THE CORRECT RESPONSE IS"
480 PRINT C$
490 W=W+1:GOTO 520
500 PRINT"RIGHT!"
510 C=C+1
520 FOR J=1 TO M-2
530 P(J-1)=P(J):NEXT
540 P(M-2)=R
550 PRINT
560 GOTO 340
580 J=256*PEEK(79)+PEEK(78)
590 J=RND(-J):RETURN
600 PRINT
610 IF C+W=0 THEN 700
```

```
620 PRINT C;" RIGHT OUT OF ";C+W
630 PRINT
640 PRINT C*100/(C+W);" PER CENT"
650 PRINT
700 PRINT"NEXT ACTION:"
710 PRINT" 1 - DRILL MORE"
720 PRINT" 2 - END PROGRAM"
730 PRINT
740 INPUT R$
750 IF R$="2" THEN END
760 IF R$="1" THEN 280
770 PRINT"ENTER 1 OR 2 PLEASE."
780 PRINT:GOTO 700
800 REM:FLASHCARD DATA FOLLOWS
810 DATA THE PEN,LA PLUMA
820 DATA THE DOOR,LA PUERTA
830 DATA THE SCHOOL,LA ESCUELA
840 DATA THE FLOOR,EL SUELO
850 DATA THE STORE,LA TIENDRA
860 DATA THE HAND,LA MANO
870 DATA THE HOUSE,LA CASA
880 DATA THE FRIEND,EL AMIGO
890 DATA THE DINNER,LA COMIDA
900 DATA THE CHAIR,LA SILLA
910 DATA TO ARRIVE,LLEGAR
920 DATA TO ASK,PREGUNTAR
930 DATA TO BUY,COMPRAR
940 DATA TO BRING,LLEVAR
950 DATA TO COME,VENIR
960 DATA TO EAT,COMER
970 DATA TO FIND,HALLAR
980 DATA TO GO,ANDAR
990 DATA TO HAVE,TENER
1000 DATA TO KNOW,SABER
1010 DATA BLUE,AZUL
1020 DATA GREEN,VERDE
1030 DATA RED,ROJO
1040 DATA WHITE,BLANCO
1050 DATA YELLOW,AMARILLO
1060 DATA ENOUGH,BASTANTE
1070 DATA FAR,LEJOS
1080 DATA FEW,POCOS
1090 DATA MANY,MUCHOS
1100 DATA NEAR,CERCA
9999 DATA XXX
```

修改程式

1. 以您自己的記憶卡來取代程式中的 DATA 敍述。以逗號來分割每張記憶卡的兩邊。在您的卡片內容中，不要出現逗號、冒號、或引號。注意保留 9999 行，因此行爲指示卡片結束的控制卡。
2. 欲改變程式對記憶卡數目的限制，可直接改130行。L爲其上限，而M爲最小值。若每張記憶卡的長度在 12 到 15 個字母之間，則目前的上限 50 將可使程式放入 4000 byte的記憶空間內（見附錄1）。若您的個人電腦有16,000 byte的可用空間，則可將記憶卡的上限訂爲400。記住不要將M訂超過 10 ，否則將會使程式的執行速度慢下來，且佔用較大的記憶空間。但M的效用可增大避免練習卡片重覆的現象。
3. 若欲使程式永遠顯示記憶卡的固定一邊（即問卡），然後學生回答另一邊（即答卡），可更改390行爲：

```
390 REM
```

欲讓程式永遠顯示另外一邊，則改爲：

```
390 GOTO 420
```

主流程

130 − 170	設定變數，建立陣列，顯示標題。
180 − 270	讀入 DATA 敘述內的記憶卡。
280 − 650	讓操作者練習背誦記憶卡。
700 − 780	顯示操作選擇與分析回答，跳到適當的程序。

主要變數

L	記憶卡數目的上限
M	記憶卡數目的最小值
R	在練習之時，隨機選中的記憶卡指標
K	輸入的記憶卡數目
W	答錯的題數
C	答對的題數
F$	含記憶卡前半邊的陣列
B$	含記憶卡後半邊的陣列
P	含最近剛顯示過的M − 1 張記憶卡指標的陣列
J	迴圈及指標變數
C$	練習時正確的答案
R$	操作者的答案。此也為工作字串變數

研究建議

1. 修改程式使能適用於教室。要求操作者練習固定的次數（譬如 20 次或 50 次），且不允許沒反應的現象。例如，您可做以下的改變：

```
440 REM
515 IF C=20 THEN 600
660 END
```

此可使程式繼續出練習題，直到答對 20 題才結束。

公制(METRIC)

學習公制系統

目　的

目前國際通用的度量衡制度爲公制。美國是個英制實行很久的國度，但也漸漸接受公制了。若您仍習慣使用英吋、英呎或以英磅來度量的話，爲了順應世界潮流，應改爲公制了。

METRIC 公制程式爲專設計讓您學習公制系統的教育程式。它以考試的形態來操作，程式以隨機的方式由資料庫組成問題。讓您比較兩個量———個以英制爲單位，一個以公制爲單位。當您答錯時，將會顯示正確的轉換式子及計算的規則。

供使用者比較的兩個量通常互爲 50% 以內。因此您往往面對相同大小領域的英制量與公制量。此可訓練您對敏銳的洞察力。

使用方法

程式首先詢問您在此次練習想做的題數，1以上的任何值都可接受。

由“執行範例”可看出問題是如何形成的。一個以英制爲單位的量和以公制爲單位的量互相比較，二者之中任意一個都可能出現於問題的前面。每個量都爲整數值。二者的相關字（“longer”長於、“hotter”熱於、“heavier”重於，等等）表示這些量的類型。

對於每個問題，有三種可能的答覆：按Y或N表示您認爲答案分別爲是（yes）或否（no）。若按其他任何鍵，表示您放棄此題。

若您答對，則值得嘉獎，程式打出一些嘉獎訊息後即進入下一個問題。若答錯或放棄此題，則會產生一些糾正文字，鼓勵同學再接再厲，且顯示此題的做法。首先印出第一個量換算成第二個量的單位之正確答案，然後印出換算的方法。每個問題的最後，程式會要求您按入任意鍵以繼續下一個題目。程式繼續產生題目，直到做完您要求的題數。結束前，它會顯示您答對的題數及答對百分比。

執行範例

```
              A METRIC QUIZ

HOW MANY QUESTIONS SHALL WE DO? 3

QUESTION 1 OF 3

IS 48 MILES LONGER THAN
   92 KILOMETERS ? ("N" key pressed)

YOU SAY 'NO'
   AND YOU'RE RIGHT - VERY GOOD!

*** HIT ANY KEY TO CONTINUE ***

QUESTION 2 OF 3

IS 73 DEGREES FAHRENHEIT HOTTER THAN
   22 DEGREES CENTIGRADE ? ("Y" key pressed)

YOU SAY 'YES'
   AND YOU'RE RIGHT - VERY GOOD!

*** HIT ANY KEY TO CONTINUE ***

QUESTION 3 OF 3

IS 79 KILOGRAMS HEAVIER THAN
   152 POUNDS ? ("N" key pressed)

YOU SAY 'NO' BUT YOU'RE WRONG

---------------------

  79 KILOGRAMS EQUALS
  174.166097 POUNDS

---- THE RULE IS ----
```

```
 1 KILOGRAM EQUALS
 2.20463 POUNDS

*** HIT ANY KEY TO CONTINUE ***

YOU GOT 2 RIGHT OUT OF 3 QUESTIONS

PERCENTAGE CORRECT = 66.6666667
```

程 式

```
100 REM: METRIC
110 REM: COPYRIGHT 1980 BY PHIL FELDMAN AND TOM RUGG
130 TEXT:Q=PEEK(78)+256*PEEK(79):Q=RND(-Q)
150 DIM ES$(30),MS$(30),R$(30),C(30),EP$(30),MP$(30)
160 B$=CHR$(32)
200 GOSUB 400:GOSUB 450
210 INPUT"HOW MANY QUESTIONS SHALL WE DO? ";NQ:NQ=INT(
    NQ):IF NQ<1 THEN 210
220 FOR J=1 TO NQ:GOSUB 600:GOSUB 900:NEXT
230 GOSUB 450:PRINT"YOU GOT ";NR;" RIGHT OUT OF ";NQ;
    " QUESTIONS":PRINT
240 P=100*NR/NQ:PRINT"PERCENTAGE CORRECT = ";P
250 END
400 RESTORE:ND=0
410 ND=ND+1:READ ES$(ND),MS$(ND),R$(ND),C(ND),EP$(ND),
    MP$(ND)
420 IF ES$(ND)<>"XXX" THEN 410
430 ND=ND-1:RETURN
450 HOME:PRINT TAB(11);"A METRIC QUIZ":PRINT:PRINT:
    RETURN
600 N=INT(ND*RND(1))+1
610 F=0:IF RND(1)>0.5 THEN F=1
620 V1=INT(RND(1)*99)+2:V3=V1*C(N):IF F=1 THEN V3=V1/C(
    N)
630 IF N=1 THEN V3=(V1-32)/1.8:IF F=1 THEN V3=(V1*1.8)+
    32
```

```
640 V2=V3*(0.5+RND(1)):V2=INT(V2+0.5):T=0:IF V2<V3 THEN
    T=1
650 GOSUB 450:PRINT"QUESTION ";J;" OF ";NQ:PRINT
660 IF F=0 THEN PRINT"IS ";V1;B$;EP$(N);B$;R$(N);" THAN"
    :PRINT B$;B$;V2;B$;MP$(N);B$;"?"
670 IF F=1 THEN PRINT"IS ";V1;B$;MP$(N);B$;R$(N);" THAN"
    :PRINT B$;B$;V2;B$;EP$(N);B$;"?"
680 GET Q$
700 IF Q$="Y" THEN PRINT:PRINT"YOU SAY 'YES' ";:R=1:
    GOTO 730
710 IF Q$="N" THEN PRINT:PRINT"YOU SAY 'NO' ";:R=0:GOTO
    730
720 PRINT:PRINT"YOU HAVE NO IDEA":R=2
730 X=T-R:IF R=2 THEN GOSUB 800:GOTO 760
740 IF X=0 THEN PRINT:PRINT
    "   AND YOU'RE RIGHT - VERY GOOD!":NR=NR+1:GOTO 760
750 PRINT"BUT YOU'RE WRONG":GOSUB 800
760 RETURN
800 PRINT:PRINT"--------------------":PRINT
810 IF F=0 THEN PRINT V1;B$;EP$(N);" EQUALS":PRINT V3;B$
    ;MP$(N)
820 IF F=1 THEN PRINT V1;B$;MP$(N);" EQUALS":PRINT V3;B$
    ;EP$(N)
830 PRINT:PRINT"---- THE RULE IS ----":PRINT
840 IF N=1 AND F=0 THEN PRINT" DEG.C = (DEG.F - 32)/1.8"
    :RETURN
850 IF N=1 AND F=1 THEN PRINT
    " DEG.F = (DEG.C * 1.8) + 32":RETURN
860 IF F=0 THEN PRINT" 1 ";ES$(N);" EQUALS":PRINT C(N);
    B$;MP$(N):RETURN
870 Q=INT(1.E5/C(N))/1.E5:PRINT" 1 ";MS$(N);" EQUALS":
    PRINT Q;B$;EP$(N):RETURN
900 PRINT:PRINT:PRINT"*** HIT ANY KEY TO CONTINUE ***"
910 GET Q$
920 RETURN
1000 DATA  DEGREE FAHRENHEIT,DEGREE CENTIGRADE,HOTTER,0.
     5
1010 DATA  DEGREES FAHRENHEIT,DEGREES CENTIGRADE
1020 DATA  MILE PER HOUR,KILOMETER PER HOUR,FASTER,1.
     60935
1030 DATA  MILES PER HOUR,KILOMETERS PER HOUR
1040 DATA  FOOT,METER,LONGER,0.3048
1050 DATA  FEET,METERS
1060 DATA  MILE,KILOMETER,LONGER,1.60935
```

```
1070 DATA  MILES,KILOMETERS
1080 DATA  INCH,CENTIMETER,LONGER,2.54
1090 DATA  INCHES,CENTIMETERS
1100 DATA  GALLON,LITRE,MORE,3.78533
1110 DATA  GALLONS,LITRES
1120 DATA  POUND,KILOGRAM,HEAVIER,0.45359
1130 DATA  POUNDS,KILOGRAMS
1999 DATA  XXX,XXX,XXX,0,XXX,XXX
```

修改程式

1. 欲使程式永遠產生固定的題數，可直接改210行設NQ爲所要之值。例如

210 NQ=10

可使程式固定產生10題。

2. 程式目前建有以下7種單位換算：

N	類型	英制	公制
1	溫度	華氏度，F	攝氏度，C
2	速度	英哩/小時	公里/小時
3	長度	英呎	公尺
4	長度	英哩	公里
5	長度	英吋	公分
6	容量	加侖	公升
7	重量	磅	公斤

若您只想針對某類型的問題做練習，在600行設N的值。因此

```
600 N=4
```

可使程式專門產生英哩和公里的問題。欲加入新的資料於程式內，可見“研究建議”部分。

3. 您也可以很容易地使程式產生的問題爲“單方向”。若要專做英制轉換到公制的題目，則使用

```
610 F=0
```

若要做公制轉換到英制的題目，則使用

```
610 F=1
```

4. 您也許希望在答對時，程式也能顯示換算的正確值及換算公式。可更改 740 行，再加入 745 行如下：

```
740 IF X=0 THEN PRINT:PRINT" AND YOU'RE RIGHT-
                            VERY GOOD"
745 IF X=0 THEN NR=NR+1:GOSUB 800:GOTO 760
```

主流程

130 － 160	設定變數及陣列
200 － 250	主要程序，操縱其他程序
400 － 430	讀入與設定資料
450	顯示標題
600 － 760	產生及顯示問題，處理使用者的回答

800 − 870	顯示正確的換算結果與換算公式
900 − 920	等待使用者按入任何鍵
1000−1999	DATA敍述，放資料庫

主要變數

ND	資料庫內的換算個數
ES$，EP$	英制單位的名稱字串陣列（單數名稱，複數名稱）
MS$，MP$	公制單位的名稱字串陣列（單數名稱，複數名稱）
R$	關係描述的字串陣列
C	單位換算係數陣列
Q	工作變數
B$	空白字元的字串常數
J	目前題號
NR	答對的題數
P	答對的百分比
NQ	一次練習的題數
N	目前問題在資料列表內的註標
F	問題的“方向”標籤（0＝英制到公制；1＝公制到英制）
V1，V2	問題的第一個量與第二個量
V3	第二個量的正確值
T	問題的正確答案標籤（1＝是；0＝否）

QS　　使用者回答的字串

R　　使用者回答標籤（0＝否，1＝是，2＝放棄）

X　　答題結果（若答對則為0）

研究建議

1. 每個建進的單位換算需要下列依序6種資料元素：

元素	資料敍述
1	英制單位（單數）
2	公制單位（單數）
3	關係描述（譬如："hotter"、"faster"等等）
4	換算係數（由英制到米制）
5	英制單位（複數）
6	公制單位（複數）

除了第四項以外，上面各項都為字串。由以上的說明當可明瞭資料列表所提供的資料情形。您可依此格式加入適當的 DATA 敍述於程式中，以擴充資料列表。新的資料 DATA 敍述可置於目前資料之後（註：1999 行以前）。1999 行為特殊的 DATA 敍述，用來標示此程式的資料結束。程式保留的空間可容納 30 項，但目前只用了7項。（注意：此格式只適合單位之間能做整

數值的換算。其中溫度並沒滿足此項限制，因此在程式內以特殊狀況來考慮。)

2. 擴充此程式，使能處理任何類型的單位換算問題。
3. 追踪被問過的問題以及答錯之題目。然後對於已被答對的問題則儘量少問。多問那些答錯的問題，以增加學生的練習次數。

數字(NUMBERS)

幫入學前兒童學習數字

目　的

此程式專為未入學的兒童所設計的教育程式。按照兒童的學習過程而言，三、四歲的小孩即能由1數到10，因此NUMBER數字程式教這些小孩來練習這些數字，寓教育於娛樂之中。

使用方法

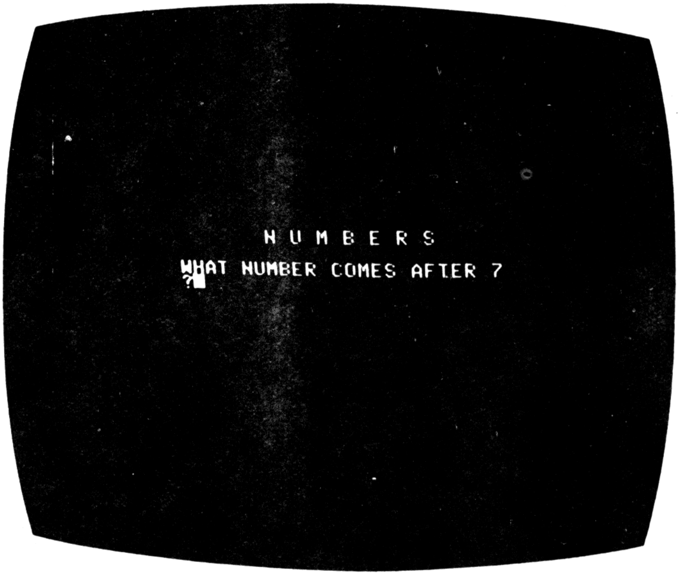
據了解，三歲的小孩就能夠學會按入讓程式執行的LOAD以及RUN鍵。因此您只要教導一下就能讓孩子自己玩電腦。程式首先問“WHAT NUMBER COMES AFTER n?”，其中n為1到9之間的數字。雖然此時小孩尚看不懂螢幕上的英文，但經教導之後就應很快看懂句尾1到9之間的數字。然後必

須回答此數字後面的數字，然後按 RETURN 鍵。

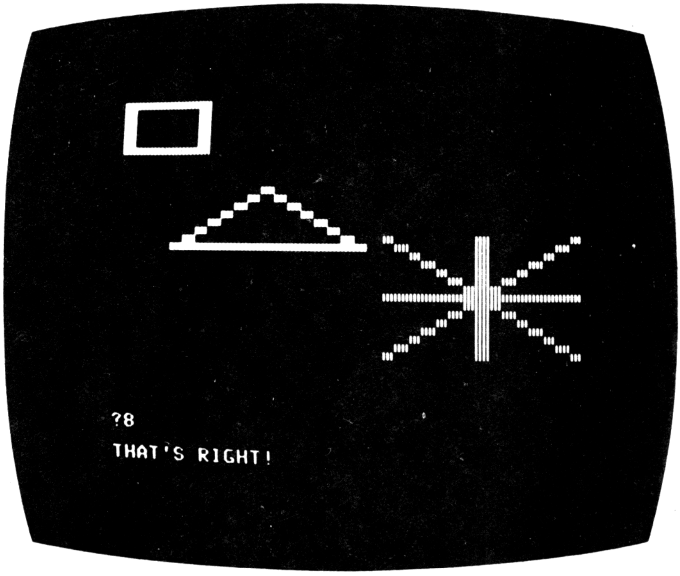
若答對，則程式顯示訊息“THAT′S RIGHT！”，暫停一、兩秒鐘，然後顯示三個幾何圖形。左上角爲長方形，中間爲三角形，然後爲一個放射形的星號顯示於螢幕的右下方。再過 5 秒後，程式清除螢幕，然後問下一個問題。相同的問題不會連續重覆問兩次，且圖形的大小是顯示之時隨機選定的。

若小朋友答錯了，會指出他的錯誤，然後相同的問題再問一遍。程式會不斷的提出問題繼續執行，除非您於回答時，打入 E 鍵代替數字，方能終止程式執行。記住，大部分的孩子耐心都有限，因此當他玩一陣子膩了以後，就不要再強迫他玩。將每節的練習弄短一點，讓小朋友們對電腦的遊戲保持一份新鮮感。

執行範例



程式問 7 之後是什麼數，等待回答。



答對了，程式顯示三個漂亮的幾何圖形，見彩色部分。

程 式

```
100 REM: NUMBERS
110 REM: COPYRIGHT 1980 BY TOM RUGG AND PHIL FELDMAN
120 GR
130 M=9:E=10:TS=11
140 HOME
150 PRINT
160 PRINT TAB(8);"N U M B E R S"
170 R=INT(M*RND(1))+1:IF R=P THEN 170
180 PRINT
190 PRINT"WHAT NUMBER COMES AFTER ";R
200 INPUT R$
210 PRINT:IF R$="E" THEN TEXT:HOME:END
220 IF VAL(R$)=R+1 THEN 300
230 PRINT"NO, THAT'S NOT IT.  TRY AGAIN."
240 GOTO 180
300 PRINT"THAT'S RIGHT!"
310 FOR X=1 TO 1000:NEXT X
320 P=R
330 E=INT(8*RND(1))+4
340 C=INT(13*RND(1))+2
400 COLOR=C:HLIN 1,E AT 1
410 HLIN 1,E AT E
420 VLIN 1,E AT 1
430 VLIN 1,E AT E
450 COLOR=C+1:FOR J=1 TO E
460 Y=TS+J:X=TS+J:PLOT X,Y:NEXT
470 FOR J=1 TO E
480 Y=TS+J:X=TS-J+2:PLOT X,Y:NEXT
490 Y=TS+E+1:FOR X=TS-E+1 TO TS+E+1
500 PLOT X,Y:NEXT
520 COLOR=INT((C+1)/2):A=28:B=25
530 FOR J=1 TO E
540 X=A+J:Y=B+J:PLOT X,Y
550 Y=B-J:PLOT X,Y
560 Y=B:PLOT X,Y
570 X=A:PLOT X,Y
580 Y=B+J:PLOT X,Y
590 Y=B-J:PLOT X,Y
```

```
600 X=A-J:PLOT X,Y
610 Y=B:PLOT X,Y
620 Y=B+J:PLOT X,Y
630 NEXT
800 FOR J=1 TO 4000:NEXT
810 GR:PRINT
820 GOTO 170
```

修改程式

1. 可改變程式詢問的問題之數字範圍，即更改130行內的M值。對於初學者，M值以3來代替原來的9；進入情況後再以5做M值，然後爲8。
2. 欲修改程式顯示“THAT'S RIGHT！”後的停頓時間，可直接改變310行內的值1000。若將此值放大兩倍，則可加倍停頓的時間。同樣地，可依同法更改800行內的4000，以改變圖形顯示後的停頓時間。
3. 若想避免顯示的圖形大小不定，可以下面敍述取代330行：

```
330 E=11
```

   若不想用11，則可使用3到11間的任意整數。
4. 設計程式依照答對題數的增加，逐漸增大顯示的圖形；若答錯，則圖形隨之縮小。可做以下的修改：

   a. 取代130行內的10值。

b. 插入此行

```
225 E=E-3:IF E<3 THEN E=3
```

c. 以下列取代330行

```
330 E=E+3:IF E>11 THEN E=11
```

主流程

120－160	設定變數，清除螢幕。
170	取由1到M的隨機整數。
180－240	詢問問題。取得答案。決定是否正確。
310	停頓大約½秒。
320－430	畫一方形。
450－500	畫一三角形。
520－630	畫一星形。
800	停頓大約5秒。
810－820	清除螢幕。跳回以問下一個問題。

主要變數

M	詢問的數字的上限
E	幾何圖形的邊長
R	由1到M之間的隨機整數

P	上次詢問的數字
R$	操作者的回答
X，Y	螢幕的位置座標
TS	三角形的開始位置（頂端）
A，B	X，Y座標值
J	註標變數

研究建議

1. 修正此程式使為詢問英文字母的下一個字母。使用 ASC 與 CHR$ 函數以選定由 A 到 Y 的字母，以及檢查答案是否正確。
2. 修改程式，使能以隨機的順序詢問由 1 到 M的每一個數字，然後於順序結束後，再重覆詢問答錯的問題。
3. 加入不同的顯示圖形。試試八角形、菱形、以及圓形。或者將此程式與圖形顯示程式組合一起。

速讀機(TACHIST)

使用速讀機增加你的閱讀速度

目 的

此程式可將您的個人電腦轉變成速讀機。速讀機可用來改變您的閱讀習慣，消除一般人閱讀的默誦習慣，以增進您的閱讀速度。程式可在螢幕上快速閃出一個字或片語，然後要您答出閃出之字。經由一番訓練後，您可發現自己可於更短的時間內讀出顯示的字。

使用方法

程式首先顯示一些簡短的介紹，然後等使用者按入任何一鍵（RESET鍵與shift 鍵除外）。若您按入RESET鍵，則程式結束執行。使用者按入一鍵後，自動消除螢幕，然後在左上角畫兩條虛綫。停頓兩秒後，螢幕上即在此二虛綫間顯示一個片語。幾秒後

，螢幕重新清除，然後問您剛才顯示的片語。

若您的回答為正確的，則下回合顯示片語的時間會減半。若您答錯，則程式指出正確的片語，且下一回合顯示片語的時間會加倍。

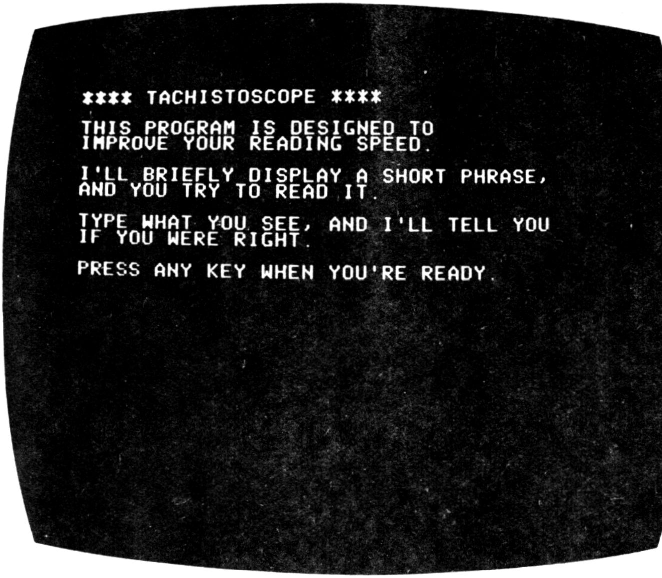
電腦最快可以 0.02 秒的時間內顯示及清除一片語。可自我測驗看自己是否能夠在這最快速度下，正確的讀出片語。

有很多專家在研究人類如何以較快的速度，與較高的理解力閱讀文章。在此我們不想深入探究這些理論（大家可自行參考目錄所列書籍），只是稍微提示一些重要的技巧，以幫助閱讀。

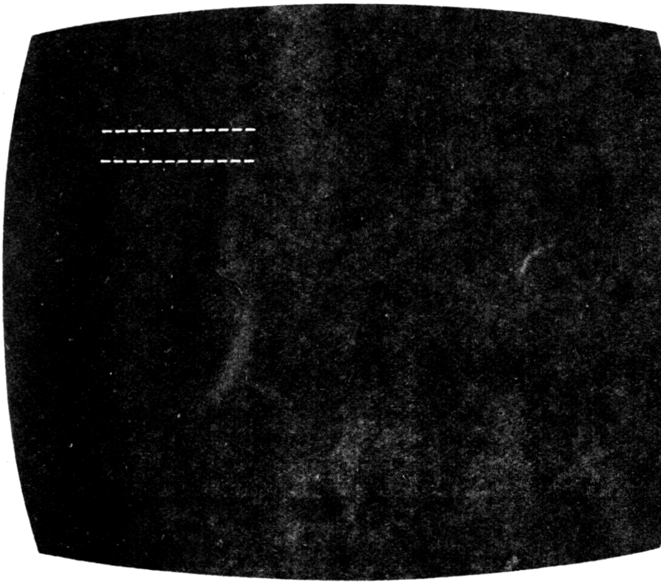
欲增快閱讀速度，您首先必須革除一次讀一個字的習慣，嚐試一口氣快速地讀整個片語。要達到一眼讀過整個片語，就必須將您的目光焦點對準此片語的中間（微偏前方），且革除口中默誦的壞習慣，如此就能夠一眼涵蓋整個片語，不須由左至右一個字一個字地讀。由於速讀機可在瞬間顯示一個片語，稍現即逝，強迫您必須在一眼的機會裡讀入整個片語，不能由左至右一字一字地唸。

若您可依此技巧修正您的閱讀習慣，且慢慢增加您能承受的片語速度，將可戲劇性地增快您的閱讀速度。

執行範例



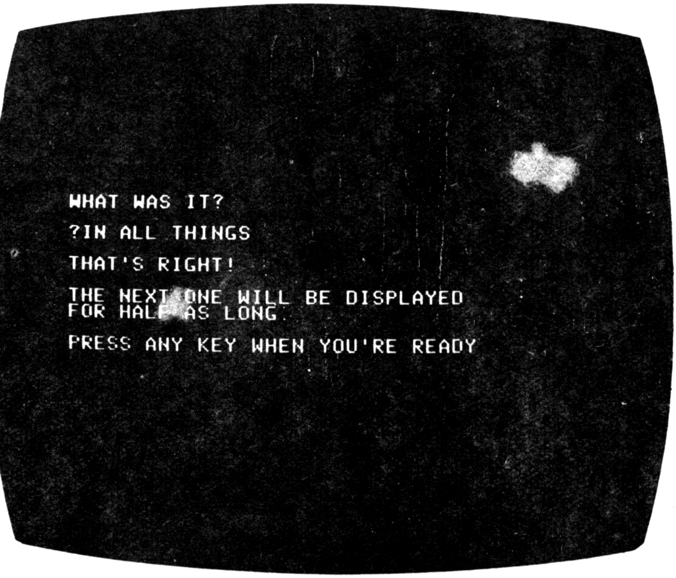
程式顯示簡介，然後等待。



程式清除螢幕，在螢幕左上角印出兩條平行綫。



程式在兩條平行綫間閃出一句片語，幾秒後，清除螢幕。



程式詢問閃出的片語，操作者回答正確，程式確認答對，指出下一個片語以一半的時間閃出。

程 式

```
100 REM: TACHISTOSCOPE
110 REM: COPYRIGHT 1980 BY TOM RUGG AND PHIL FELDMAN
130 T=256
140 L=50
150 DIM T$(L)
160 C=0
170 READ R$
180 IF R$="XXX" THEN 250
190 C=C+1
200 IF C>L THEN PRINT"TOO MANY DATA STATEMENTS":END
210 T$(C)=R$
220 GOTO 170
250 TEXT:HOME:GOSUB 800
270 PRINT"**** TACHISTOSCOPE ****"
280 PRINT
290 PRINT"THIS PROGRAM IS DESIGNED TO"
300 PRINT"IMPROVE YOUR READING SPEED."
310 PRINT
320 PRINT"I'LL BRIEFLY DISPLAY A SHORT PHRASE,"
330 PRINT"AND YOU TRY TO READ IT."
340 PRINT
350 PRINT"TYPE WHAT YOU SEE, AND I'LL TELL YOU"
360 PRINT"IF YOU WERE RIGHT."
370 PRINT
410 PRINT"PRESS ANY KEY WHEN YOU'RE READY."
420 GET R$:IF ASC(R$)=27 THEN END
430 R=INT(C*RND(1))+1
440 IF R=P1 OR R=P2 OR R=P3 THEN 430
450 IF R=P4 OR R=P5 THEN 430
460 GOSUB 840:FOR K=1 TO 1500:NEXT
470 VTAB 2:PRINT T$(R)
480 FOR J=1 TO T:NEXT
500 HOME:FOR K=1 TO 500:NEXT
510 PRINT:PRINT:PRINT:PRINT
520 PRINT"WHAT WAS IT?"
530 PRINT:INPUT R$
550 IF R$<>T$(R) THEN 700
560 PRINT:PRINT"THAT'S RIGHT!"
570 T=T/2
```

```
590 R$="FOR HALF AS LONG."
600 P1=P2:P2=P3:P3=P4:P4=P5:P5=R
610 PRINT
620 IF T<=16 THEN T=16:R$="AT MAXIMUM SPEED."
630 PRINT"THE NEXT ONE WILL BE DISPLAYED":PRINT R$
640 PRINT:GOTO 410
700 PRINT"NO, THAT'S NOT IT.  IT WAS"
710 PRINT:PRINT"'";T$(R);"'"
720 T=T*2
730 IF T>2048 THEN T=2048:R$="AT THE SAME SPEED.":GOTO
    600
740 R$="FOR TWICE AS LONG.":GOTO 600
800 J=256*PEEK(79)+PEEK(78)
810 J=RND(-J):RETURN
840 HOME:PRINT"------------"
850 PRINT
860 PRINT"------------"
870 RETURN
910 DATA"AT THE TIME"
920 DATA"THE BROWN COW"
930 DATA"LOOK AT THAT"
940 DATA"IN THE HOUSE"
950 DATA"THIS IS MINE"
960 DATA"SHE SAID SO"
970 DATA"THE BABY CRIED"
980 DATA"TO THE STORE"
990 DATA"READING IS FUN"
1000 DATA"HE GOES FAST"
1010 DATA"IN ALL THINGS"
1020 DATA"GREEN GRASS"
1030 DATA"TWO BIRDS FLY"
1040 DATA"LATE LAST NIGHT"
1050 DATA"THEY ARE HOME"
1060 DATA"ON THE PHONE"
1070 DATA"THROUGH A DOOR"
1080 DATA"WE CAN TRY"
1090 DATA"MY FOOT HURTS"
1100 DATA"HAPPY NEW YEAR"
9999 DATA  XXX
```

修改程式

1. 可更改螢幕顯示出來的片語，只需改變程式內由 910 行起的 DATA 敍述即可。可加入新的片語或更改原先的片語。另外程式內140行必須設定 DATA 敍述的最大範圍。因此，若允許100個 DATA 敍述，可更改 140 行為

```
140 L=100
```

必須確使您新輸入的 DATA 敍述的格式與程式內原來的 DATA 敍述格式相同。開始練習速讀時，您也許必須由較短的片語或單字開始訓練，然後再漸漸地加長片語的長度。注意不要更改 9999 行，此行為指示 DATA 敍述的最後一行之特殊用途。若您有 4000 byte可用記憶空間（見附錄1），您大概可容納 80 個片語於程式之內。

2. 欲更改第一個片語顯示的時間長度，可直接改變 130行內的T值。若T值加倍，則顯示的時間加倍。注意，莫使T值小於 16 。

3. 欲使所有的片語都以相同的時間長度顯像。可刪除 570 行與 720 行，然後插入以下幾行：

```
595 R$="AT THE SAME SPEED"
725 R$="AT THE SAME SPEED":GOTO 600
```

4. 若您想改變片語顯像前的停頓時間，可更改 460 行內的 1500 值。若欲停頓 5 秒鐘，可改變爲 4000 ，欲停頓 1 秒，則改爲 800 。

5. 若想在程式內添加一種記憶卡目錄狀態，顯示出所有的片語，而不須回答這些片語。可直接插入以下三行：

```
515 GOTO 710
595 R$= "AT THE SAME SPEED"
715 GOTO 590
```

如此即能顯示每一片語（而且顯像的時間都相同），然後若您想確定此片語，可使它重新顯像。

主流程

130 － 150	設定變數。
160 － 220	將 DATA 敍述讀入 T$ 陣列。
250 － 370	顯示簡介。
410 － 420	等待操作者按入一鍵。
430 － 450	由 T$ 陣列隨機選一個片語，必須確使其不與前面 5 個已顯像的片語相同。
460	清除螢幕，然後顯示兩條水平綫。
470 － 500	依適當的時間長度顯示片語。
510 － 530	詢問所顯像的片語。
550	判斷操作者按入的答案是否與顯像的片語相同。

560－640	若答案正確，則縮短下一個片語的顯像時間。儲存片語註標以免重覆。跳回等待操作者按入一鍵。
700－740	答錯，顯示正確的片語，增長下一個片語的顯像時間，確使時間長度不超過最大極限。
800－810	副程式，設定 RND 隨機數函數。
840－870	副程式，顯示水平的兩條虛綫。
910－9999	欲顯示片語的DATA敍述。

主要變數

T	片語顯像的時間長度
J	迴圈變數
L	片語數目的範圍
T$	片語陣列（讀自DATA敍述）
C	實際讀入的片語數目之計數器
R$	暫時字串變數。而且，做爲操作者的回答字串
R	工作變數。而且，做爲欲顯像片語的註標
P1，P2，P3，P4，P5	前面5個片語的註標
K	暫時工作變數

研究建議

1. 若不要以隨機的方式選擇片語，而以循序方式顯示整個片語列表。可更改250行，設R爲0，在430行使R加1，然後檢查R是否大於C。
2. 程式目前只是確使顯示的片語不與前五個片語重覆。請修改程式，使能夠避免與前十個（或更多）片語重覆。必須在440行、450行與600行做一些修改。
3. 請記錄答對與答錯的題數，然後顯示出每次練習的答對比率。或則，進而依照答對比率加上顯像時間，甚至考慮片語的困難度來計算答題的分數。
4. 爲此程式添加新的能力，使其能夠附上數字顯像狀態，隨機地選出顯示二位數到七位數的數字，讓操作者練習。若操作者先練習數字顯像，可使片語顯像練習更容易進入情況。

字彙(VOCAB)

擴充你的字彙

目 的

是不是常苦惱又遺忘了一些單字？不要緊，此字彙程式可提供您做自我訓練與加強記憶一些學科的術語和英文單字的測驗。它可讓您依自己的學習能力與速度做練習，不必擔心別的同學會笑您，老師會責罵您。當您發生一些難堪的錯誤時，只有電腦曉得，它會悄悄地提醒您，不會告訴別人。您可很容易地更換不同的字彙列表，使程式具有多種版本，教您不同領域的字彙。

使用方法

由執行範例就可很明瞭地看出此程式的使用情形。當您按入"RUN"後，程式立即詢問您想練習的題

數。若您回答五題以內，您仍須做五個題目，否則，就按照您所給的題數來做練習。

接著，程式就會提出一系列的選擇題。題目可分爲兩類——給定一個單字，要您從其下選出一個正確的定義；或者給定一個定義，要您其下所列的單字選出一個正確的單字。這兩類題目都依隨機選出。若您答對，程式會告訴您。若答錯，則顯示正確的答案。接著顯示您到目前爲止答對的題數與已答的題數。

當做完最後一題後，程式會顯出您答對的比率分數以及對您的表現所做的評語。然後您可選擇做下一回合的練習或者休息。

執行範例

```
RUN

**** VOCABULARY QUIZ ****

THIS PROGRAM WILL TEST YOUR KNOWLEDGE
OF SOME USEFUL VOCABULARY WORDS.

HOW MANY QUESTIONS SHALL WE DO? 5


 1 -- WHAT WORD MEANS ALL-KNOWING?
        1 -- LACONIC
        2 -- HEDONISTIC
        3 -- OMINOUS
        4 -- CONGENITAL
        5 -- OMNISCIENT
```

```
? 5
RIGHT!

 2 -- WHAT DOES PARSIMONIOUS MEAN?
      1 -- INDIFFERENT OR UNINTERESTED
      2 -- KEEN IN JUDGMENT
      3 -- STINGY OR FRUGAL
      4 -- WEAK OR EXHAUSTED
      5 -- OF UNKNOWN OR HIDDEN ORIGIN

? 4
NO, THE ANSWER IS NUMBER 3

(. . . later)

YOU HAVE 3 RIGHT OUT OF 5 QUESTIONS.

THAT'S 60 PERCENT.
NOT BAD, BUT ROOM FOR IMPROVEMENT.

WANT TO TRY AGAIN?NO

CHECK YOU LATER.
```

程 式

```
100 REM: VOCABULARY QUIZ
110 REM: COPYRIGHT 1980 BY TOM RUGG AND PHIL FELDMAN
300 GOSUB 1000
400 GOSUB 2000
500 GOSUB 3000
600 GOSUB 4000
700 GOSUB 5000
800 GOSUB 6000
900 IF E=0 THEN 500
910 GOTO 300
990 REM
1000 IF E<>0 THEN 1060
1010 TEXT:HOME
1020 PRINT"**** VOCABULARY QUIZ ****"
1030 PRINT
```

```
1040 PRINT"THIS PROGRAM WILL TEST YOUR KNOWLEDGE"
1050 PRINT"OF SOME USEFUL VOCABULARY WORDS."
1060 PRINT
1110 INPUT"HOW MANY QUESTIONS SHALL WE DO? ";L
1120 IF L>4 THEN 1140
1130 PRINT"THAT'S NOT ENOUGH.  LET'S DO 5.":L=5
1140 IF E<>0 THEN 1200
1150 PRINT
1170 J=256*PEEK(79)+PEEK(78)
1180 J=RND(-J)
1200 RETURN
2000 IF E<>0 THEN 2200
2010 C=5
2020 D=26
2030 DIM D$(D),E$(D)
2040 DIM P(C)
2050 J=1
2060 READ D$(J)
2070 IF D$(J)="XXX" THEN 2140
2090 READ E$(J)
2100 J=J+1
2110 IF J<=D THEN 2060
2120 PRINT"TOO MANY DATA STATEMENTS."
2130 PRINT"ONLY FIRST ";D;" ARE USED."
2140 D=J-1
2200 Q=1
2210 E=0
2220 Q1=0
2300 RETURN
3000 FOR J=1 TO C
3010 P(J)=0
3020 NEXT
3030 FOR J=1 TO C
3040 P=INT(D*RND(1))+1
3045 IF P=P1 OR P=P2 OR P=P3 THEN 3040
3050 FOR K=1 TO J
3060 IF P(K)=P THEN 3040
3070 NEXT K
3080 P(J)=P
3090 NEXT J
3110 A=INT(C*RND(1))+1
3200 RETURN
4000 PRINT
4010 M=RND(1)
```

```
4020 IF M>.5 THEN 4100
4030 PRINT Q;" -- WHAT WORD MEANS ";E$(P(A));"?"
4040 FOR J=1 TO C
4050 PRINT TAB(5);J;" -- ";D$(P(J))
4060 NEXT
4070 GOTO 4210
4100 PRINT Q;" -- WHAT DOES ";D$(P(A));" MEAN?"
4110 FOR J=1 TO C
4120 PRINT TAB(5);J;" -- ";E$(P(J))
4130 NEXT
4210 RETURN
5000 INPUT R
5010 IF R>=1 AND R<=C THEN 5050
5020 PRINT"I NEED A NUMBER FROM 1 TO ";C
5030 GOTO 5000
5050 IF (R=A) THEN 5100
5060 PRINT"NO, THE ANSWER IS NUMBER ";A
5070 GOTO 5210
5100 PRINT"RIGHT!"
5110 Q1=Q1+1
5210 IF Q=1 THEN 5300
5220 PRINT"YOU HAVE ";Q1;" RIGHT OUT OF ";Q;
     " QUESTIONS."
5300 P3=P2
5310 P2=P1
5320 P1=P(A)
5330 RETURN
6000 Q=Q+1
6010 IF Q<=L THEN RETURN
6020 E=1
6030 Q=Q1*100/(Q-1)
6040 IF Q>0 THEN 6070
6050 PRINT"WELL, THAT'S A 'PERFECT' SCORE..."
6060 GOTO 6200
6070 PRINT"THAT'S ";Q;" PERCENT."
6080 IF Q>25 THEN 6110
6090 PRINT"CONGRATULATIONS ON AVOIDING A SHUTOUT."
6100 GOTO 6200
6110 IF Q>50 THEN 6140
6120 PRINT"YOU CAN USE SOME MORE PRACTICE."
6130 GOTO 6200
6140 IF Q>75 THEN 6170
6150 PRINT"NOT BAD, BUT ROOM FOR IMPROVEMENT."
6160 GOTO 6200
6170 PRINT"VERY GOOD!"
```

```
6180 IF Q>95 THEN PRINT"YOU'RE ALMOST AS SMART AS I AM!"
6200 PRINT
6210 INPUT"WANT TO TRY AGAIN? ";R$
6220 IF LEFT$(R$,1)<>"N" THEN 6240
6230 PRINT:PRINT"CHECK YOU LATER.":PRINT:END
6240 IF LEFT$(R$,1)<>"Y" THEN 6200
6250 RETURN
7000 REM: ON LINE 2020, D MUST BE AT LEAST ONE GREATER
7005 REM: THAN THE NUMBER OF DIFFERENT WORDS.
7010 DATA  ANONYMOUS,"OF UNKNOWN OR HIDDEN ORIGIN"
7020 DATA  OMINOUS,"THREATENING OF MENACING"
7030 DATA  AFFLUENT,"WEALTHY"
7040 DATA  APATHETIC,"INDIFFERENT OR UNINTERESTED"
7050 DATA  LACONIC,"TERSE"
7060 DATA  INTREPID,"FEARLESS OR COURAGEOUS"
7070 DATA  GREGARIOUS,"SOCIAL OR COMPANY-LOVING"
7080 DATA  ENERVATED,"WEAK OR EXHAUSTED"
7090 DATA  VENERABLE,"WORTHY OF RESPECT OR REVERENCE"
7100 DATA  DISPARATE,"DIFFERENT AND DISTINCT"
7110 DATA  VIVACIOUS,"LIVELY OR SPIRITED"
7120 DATA  ASTUTE,"KEEN IN JUDGMENT"
7130 DATA  URSINE,"BEARLIKE"
7140 DATA  PARSIMONIOUS,"STINGY OR FRUGAL"
7150 DATA  OMNISCIENT,"ALL-KNOWING"
7999 DATA  XXX
```

修改程式

1. 可依您自己的需要，加入新的 DATA 敍述於 7000 行與 7999 行之間，或者更改原來的 DATA 敍述。應留意避免使用兩個或多個單字具有相同或類似的定義，因程式可能同時選到它們，而出現於同一個題目內造成多種答案的現象。注意，每個 DATA 敍述首先為單字，然後為逗點，然後為一個定義或同義字。必須確使定義內不出現

逗點或冒號（若無法避免，就以引號括在定義的外面）。若您加入新的 DATA 敍述，就必須修正 2020 行的 D 值，使其至少比所有的單字多 1。程式內所能容納的 DATA 敍述個數，依每個敍述的長度與您的電腦可用的記憶容量而定。若以目前程式內的 DATA 敍述的平均長度來算的話，假如您有 4000 byte 的可用空間（見附錄 1），則可容納 40 個單字及其定義，而若有 10,000 byte的可用空間，則可容納 200 個單字以上。必須保留 7999 行的敍述，因它用來指示 DATA 敍述的結束。

2. 若您不想做五選一的題目，希望做三選一或四選一，可直接更改 2010 行內的 C 值。
3. 若您想固定每回合練習的題數，不想讓使用者自行設定，可刪除 1110 行到 1140 行，然後插入以下幾行：

```
1110 PRINT"WE'LL DO TEN QUESTIONS."
1120 L=10
```

此可使每回合做 10 個題目。當然，您也可以使用別的數目。

主流程

300 − 910　主要程序，呼叫各個副程式。

1000−1200	顯示簡介。決定此回合的題數。
2000−2300	讀入字彙單字及其定義於陣列內，處理維護工作。
3000−3200	選定一個題目的各個選擇答案，且決定何者爲正確答案。
4000−4210	決定題目的類型，然後顯示題目。
5000−5330	接受操作者的答案。判斷其是否正確，記錄分數，存最近三個正確答案的註標。
6000−6250	顯示測驗的分數。詢問操作者下一個動作。
7000−7999	含字彙單字與定義的DATA敍述。

主要變數

E	設爲1以避免在第一回合後重覆地顯示簡介
L	詢問的題數上限
R	操作者回答的答案
C	每個題目的選擇項個數
D	DATA 敍述的個數上限，使用於 DIM 陣列，至少比 DATA 敍述的個數多1。
D$	字彙單字陣列
E$	定義陣列
P	每個問題的可能答案之陣列
J	工作變數（FOR-NEXT迴圈的指標）
Q	迄今已詢問的題數（用來計算答對的比率）

Q1 迄今答對的題數
P 工作變數
P1, P2, P3 最近三個題目的正確答案
A 正確答案在P陣列的註標
M 決定題目類型的工作變數
R$ 操作者於一回合後的是否繼續之答覆

研究建議

1. 修改 6030 行到 6200 行，依照答對的比率分數，來顯示適切的評語。例如，若答對爲 100%，則顯示贊賞佳許的訊息，若爲 95 到 99 ，則另一種訊息，等等。
2. 在簡介之時，詢問操作者的姓名，然後在其後所顯示的訊息內，適當地加入他或她個人的稱呼。
3. 目前程式只檢查最近的三個問題以避免重覆，請改爲檢查最近的八個或十個問題。（可修改 3045 行及 5300 行到 5320 行。）
4. 記錄追踪操作者答錯的問題。然後經過幾條問題後，再重新詢問這些答錯的問題。

第三部分：遊戲程式：

和電腦一起玩耍

導 論

每個人都有幾分未泯的童心，喜歡玩遊戲。用你的Apple 個人電腦來玩電腦遊戲。不但有趣，娛樂性極高，而且可鬆弛神經，訓練思考力。益智的遊戲通常可訓練你作有系統的思考，預做打算或者有條理的思惟。電腦遊戲也是治療一般人們的“恐電腦症”最好的方法。此處收集了一些適合各年紀各階層人們的口味之遊戲。

也許你喜歡有挑戰性技術性的遊戲，像西洋棋之類。WARI瓦利棋遊戲完全不靠運氣，必須要有嚴密的思考方能得勝。電腦扮演你的對手，是個很堅強的對手。

也許你喜歡快速反應，具有刺激性的遊戲。GRO-AN 呻吟的骰子遊戲爲一種快速的骰子遊戲，其大部分靠運氣及直覺，不需什麼技術。電腦隨時等待你的挑戰。

JOT 猜字遊戲是互猜秘密字遊戲，你與電腦各

自取一個秘密字，然後較量心機，猜出對方的秘密字。

你是否對謎題特別感興趣？若果眞如此，可試一下DECODE 猜碼遊戲，電腦選定一個密碼，然後要你解碼。

圖形競賽遊戲爲最近幾年逐漸盛行的遊戲。我們收集了兩種此類的遊戲。ROADRACE瘋狂大賽車遊戲將你置於高速的賽車前輪之後，你必須小心地控制方向盤，以免撞出公路外。OBSTACLE阻礙賽遊戲可讓你及你的朋友一道玩，在遊戲內互相攻擊阻礙。你們每個人都必須避免通過別人所鋪的道路，以及自己的道路。

猜碼(DECODE)

猜電腦的密碼

目　的

猜碼遊戲是個類似謎題的遊戲。你可以與朋友比賽，看誰最先解出謎題。每當你玩此遊戲時，就面臨一個新的謎題。

此遊戲讓你猜電腦選定的密碼，猜的次數越少表示你的功力越高。程式會提供你每次猜測的準確情形。依據電腦提供的資料，小心地選擇你所猜的數值，就可發現你可在 5 次或 6 次之內猜得正確的密碼。

也許你剛開始嚐試時，必須猜較多次，但只要經過練習後，就可由以前累積的經驗獲得解碼的技巧。

使用方法

程式首先顯示遊戲玩法簡介，其大意如下：

程式選定一個密碼讓你猜。此密碼爲由數字 1 到 6 組成的四位數，譬如，電腦也許選定 6153 或 2242爲密碼。

你的責任是在最少的次數之下猜中此一密碼。每當你猜一次後，程式即告訴你幾個“黑數”及幾個“白數”。黑數代表猜對的位數個數——數字對且位置也對。因此，若密碼爲 6153 ，而你猜 4143，程式會告訴你黑數爲 2（因爲 1 和 3 爲正確）。當然，不能告訴你確實是那一位數正確這必須要靠你的聰明才智加上別次猜測所得的資料判斷得到。

白數則表示你所猜的數值內的一位數的數字對但位置不對。例如，若密碼爲 6153 而你猜的數值爲 1434，則程式會告訴你白數爲 2 ，因爲 1 與 3 的數字對，但位置錯。

被算入黑數的位數，就不計入白數內，而且每個位數只能算一次，不能同時列入白數與黑數。如此才不易混淆，尤其是密碼或你猜測的數值有位數值重覆的現象時。例如，若密碼爲 1234 ，而你猜測的值爲 4444時，則只有黑數 1 ，並無白數。若密碼爲 2244 而你猜測的值爲 4122 ，則無黑數，白數爲 3 。

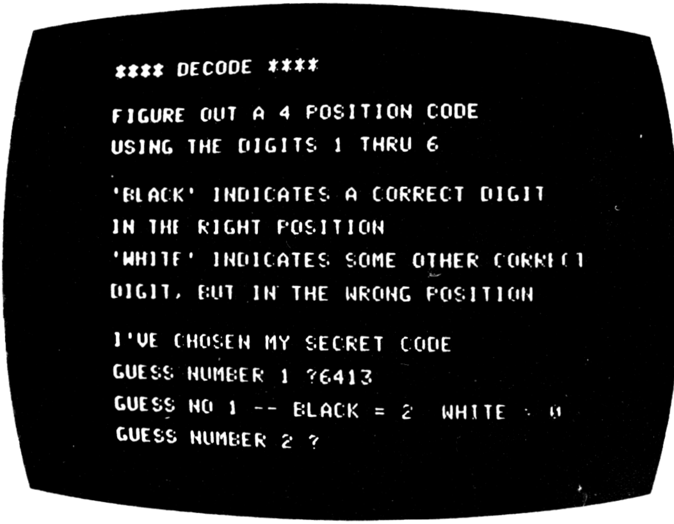
也許你剛開始對於黑數與白數的計算方式不太習慣，相信經幾次練習後，你即能學得其竅門。

在遊戲進行之中，你可隨時按入 S 鍵代表“SUMMARY”總結。程式即能清除螢幕，顯示迄今各猜測值及其所得黑數與白數。

也許，你的毅力不夠，不想再猜，宣佈放棄，則只

要按入Q鍵（表示“quit”放棄），就能看到你百思不解的正確密碼。或者，你百折不撓，一直猜下去，終於獲得正確答案（黑數為4，白數為0），或者你的猜測次數已超過上限——12次，程式即宣佈答案。

執行範例



程式顯示簡介，選定一密碼，要操作者猜第一次。操作者猜後，程式回答其黑數及白數，然後要操作者猜第二次。

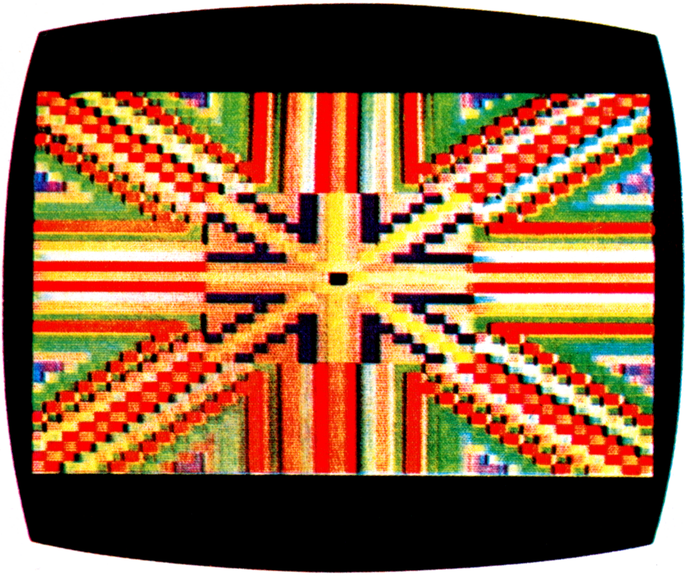


不久後，在同次猜碼中，操作者詢問總結。然後再猜出正確答案。程式確認猜對後，詢問是否再玩。

程　式

```
100 REM: DECODE
110 REM: COPYRIGHT 1980 BY TOM RUGG AND PHIL FELDMAN
120 D=6:P=4:L=12
130 DIM G$(L),G(P),C(P),B(L),W(L)
150 GOSUB 1200
170 GOSUB 300:GOSUB 370
180 PRINT"GUESS NUMBER ";G;" ";
190 INPUT A$
200 IF LEFT$(A$,1)="S" THEN 500
210 IF LEFT$(A$,1)="Q" THEN 600
220 GOSUB 700:IF K=1 THEN 180
230 GOSUB 800
240 GOSUB 1000
250 IF B(G)=P THEN 2000
260 G$(G)=A$
270 G=G+1:IF G>L THEN 2200
280 GOTO 180
300 G=1:C$=""
310 RETURN
370 FOR J=1 TO P
380 R=INT(D*RND(1))+1
390 C$=C$+STR$(R)
400 NEXT J
410 PRINT"I'VE CHOSEN MY SECRET CODE."
420 PRINT
430 RETURN
500 IF G=1 THEN PRINT"NO GUESSES YET.":GOTO 180
510 HOME:PRINT"SUMMARY":PRINT
520 PRINT"NO.   GUESS   BLACK   WHITE"
530 PRINT:FOR J=1 TO G-1
540 PRINT J;TAB(7);G$(J);TAB(17);B(J);TAB(25);W(J)
550 IF G<10 THEN PRINT
560 NEXT:PRINT
570 GOTO 180
600 PRINT
610 PRINT"CAN'T TAKE IT, HUH?"
620 PRINT:PRINT"WELL, MY CODE WAS ";
630 FOR J=1 TO 4
640 PRINT" .";
650 FOR K=1 TO 900:NEXT
660 NEXT J
670 PRINT C$:PRINT
680 GOTO 2090
```

彩色部分



萬花筒


GROAN
HOW MUCH NEEDED TO WIN
(BETWEEN 50-100 IS BEST) ? 75

呻吟的骰子


GROAN
I GET FIRST ROLL

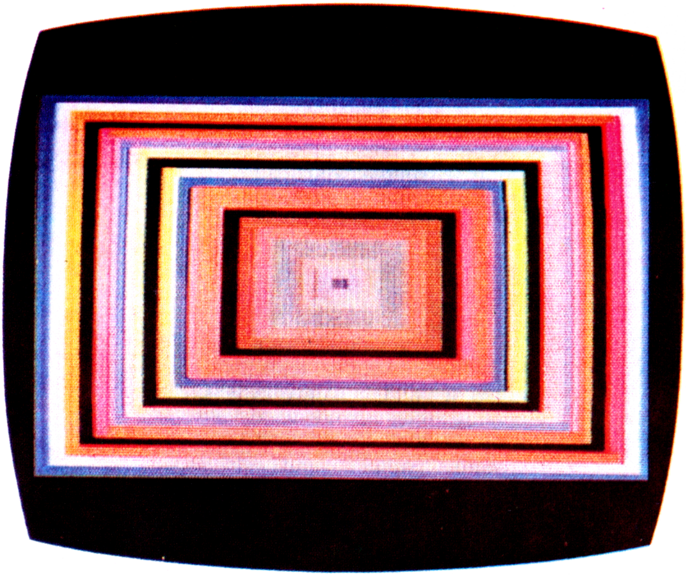
呻吟的骰子



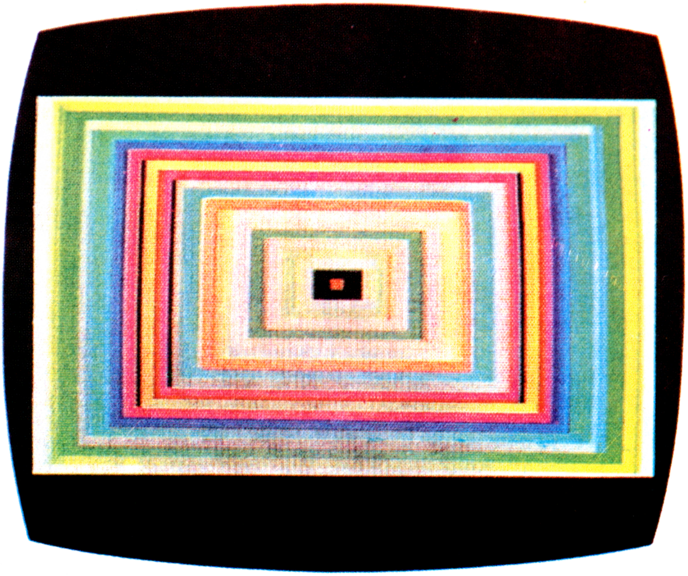
呻吟的骰子



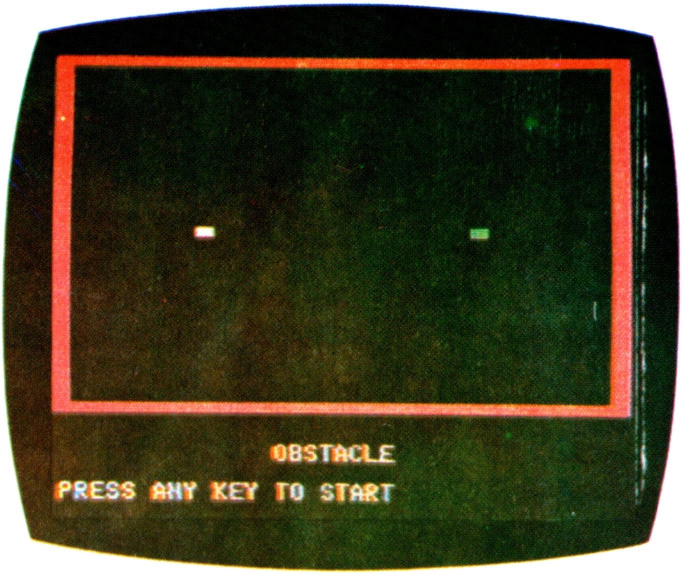
數　字



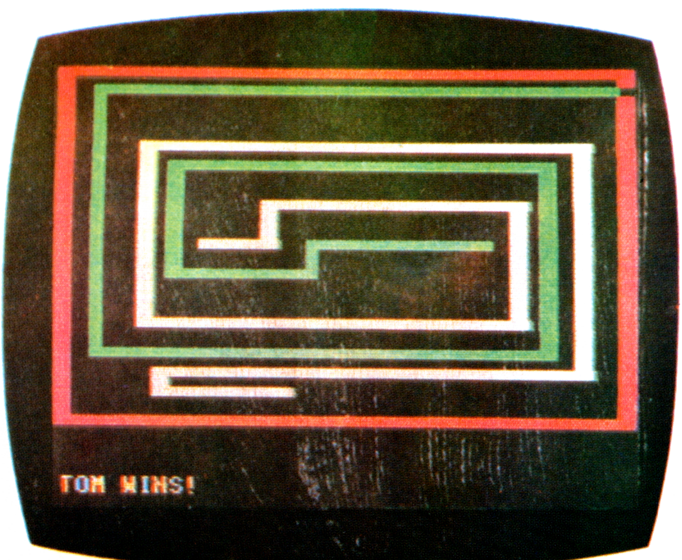
同心圖



同心圖



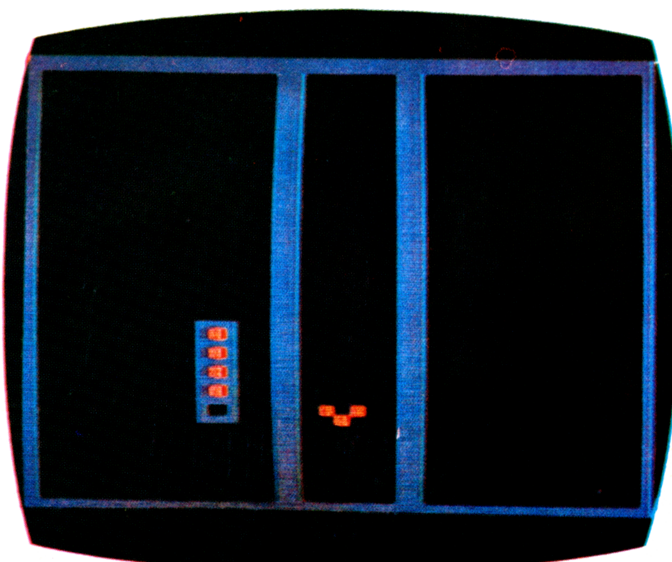
障礙賽



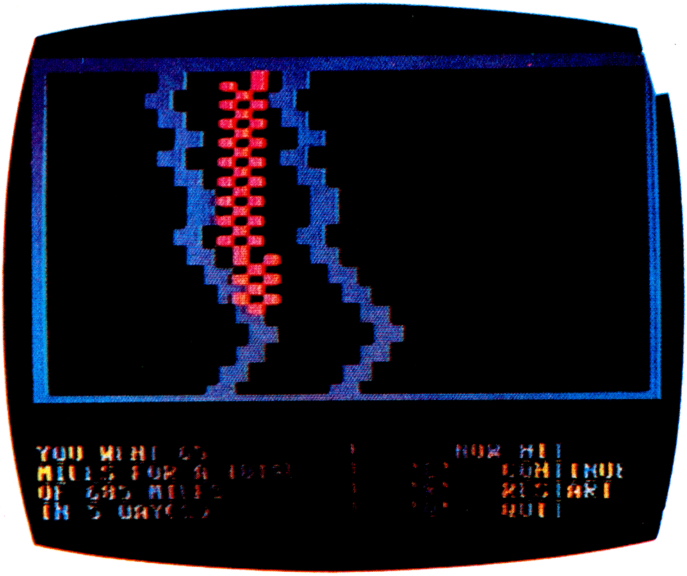
障礙賽



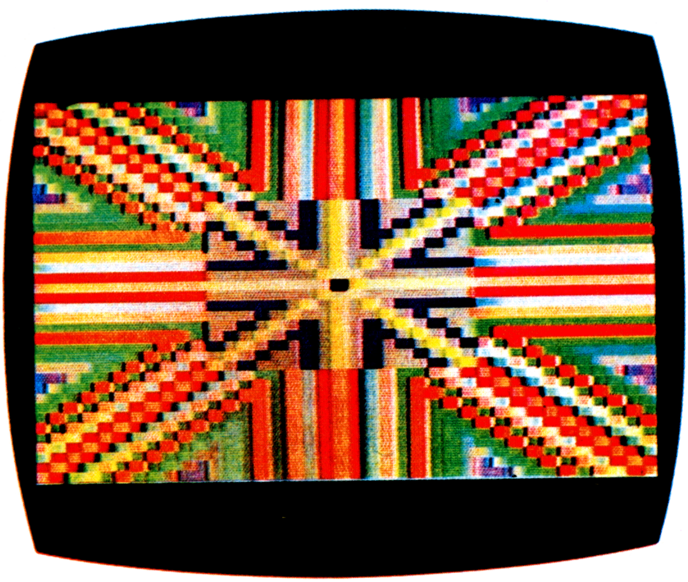
瘋狂大賽車



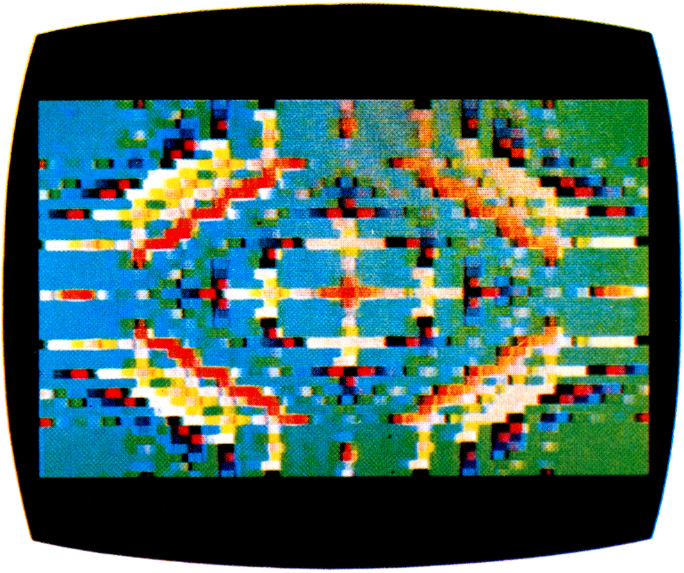
瘋狂大賽車



瘋狂大賽車



萬花筒



星光閃爍



星光閃爍

```
700 K=0:IF LEN(A$)<>P THEN 780
710 FOR J=1 TO P
720 R=VAL(MID$(A$,J,1))
730 IF R<1 OR R>D THEN 780
740 NEXT
750 RETURN
780 PRINT"ILLEGAL.  TRY AGAIN."
790 K=1:RETURN
800 B=0:W=0
810 FOR J=1 TO P
820 G(J)=VAL(MID$(A$,J,1))
830 C(J)=VAL(MID$(C$,J,1))
840 IF G(J)=C(J) THEN B=B+1:G(J)=0:C(J)=0
850 NEXT
860 FOR J=1 TO P:IF C(J)=0 THEN 920
870 H=0:FOR K=1 TO P
880 IF C(J)=0 THEN 910
890 IF C(J)<>G(K) THEN 910
900 H=1:G(K)=0:C(J)=0
910 NEXT K:W=W+H
920 NEXT J
930 RETURN
1000 B(G)=B:W(G)=W:PRINT
1010 PRINT"GUESS NO.";G;" -- BLACK = ";B;"  WHITE = ";W
1020 PRINT:RETURN
1200 TEXT:HOME
1210 PRINT"**** DECODE ****"
1220 PRINT:PRINT
1230 PRINT"FIGURE OUT A ";P;" POSITION CODE"
1240 PRINT
1250 PRINT"USING THE DIGITS 1 THRU ";D
1260 PRINT:PRINT
1270 PRINT"'BLACK' INDICATES A CORRECT DIGIT"
1280 PRINT:PRINT"IN THE RIGHT POSITION."
1290 PRINT
1300 PRINT"'WHITE' INDICATES SOME OTHER CORRECT"
1310 PRINT
1320 PRINT"DIGIT, BUT IN THE WRONG POSITION."
1330 PRINT:PRINT
1340 J=256*PEEK(79)+PEEK(78)
1350 J=RND(-J):RETURN
2000 PRINT
2010 PRINT"YOU GOT IT IN ";G;" GUESSES."
2020 IF G<5 THEN B$="OUTSTANDING!"
2030 IF G=5 OR G=6 THEN B$="PRETTY GOOD"
```

```
2040 IF G=7 THEN B$="NOT BAD"
2050 IF G=8 THEN B$="NOT TOO GREAT"
2060 IF G>8 THEN B$="PRETTY BAD"
2070 PRINT:PRINT"...THAT'S ";B$
2080 PRINT
2090 INPUT"WANT TO TRY AGAIN? ";A$
2100 IF LEFT$(A$,1)="Y" THEN 150
2110 IF LEFT$(A$,1)<>"N" THEN 2090
2120 PRINT:PRINT"COWARD.":PRINT
2130 END
2200 PRINT
2210 PRINT"THAT'S YOUR LIMIT OF ";L;" GUESSES."
2220 PRINT
2230 PRINT"MY CODE WAS ";C$
2240 GOTO 2080
```

修改程式

1. 可修改程式120行，以改變密碼的複雜程度。例如，下面一行所示，可允許使用者做 15 次的猜測，密碼共有五位數，每位數的數字由 1 到 8：

   120 D = 8 : P = 5 : L = 15

   如此修改以後，顯示的遊戲簡介即能自動地反映出 D 與 P 的新值。注意，D 或 P 都不能大於 P。

2. 可修改程式，使其於每次猜測後，都能自動地顯示總結（“ Summary ”）的資料。直流修改 280 行如下：

   280 GOTO 500

主流程

120-170	設定變數。顯示遊戲簡介。選定密碼。
180-240	輸入操作者的猜測。分析此猜測。顯示結果。
250	判斷操作者是否猜對。
260-280	儲存此猜測。猜測計數器加1。判斷猜測的次數是否已超過。
300-310	副程式，設定變數。
370-430	副程式，選定密碼及通知操作者。
500-570	副程式，顯示迄今的猜測總結。
600-680	副程式，當操作者宣佈放棄後，慢慢地顯示密碼。
700-790	副程式，判斷操作者的猜測數值是否合法。
800-930	副程式，判斷猜測值的黑數及白數。
1000-1020	副程式，顯示猜測值的黑數及白數。
1200-1340	副程式，顯示標題及遊戲簡介。
2000-2130	副程式，操作者猜出密碼後，分析其表現狀況，然後詢問是否再玩。
2200-2240	副程式，操作者猜測次數超過上限後，顯示密碼。

主要變數

D	密碼的位數可能數字（註：由 1 到D ）
P	密碼的位數個數
L	猜測次數的上限
G $	儲存各個猜測的陣列
G , C	每個猜測值於分析時的工作變數
B , W	每次猜測所得的黑數與白數的陣列
R , H	工作變數
G	猜測次數計數器
A $	操作者的回答
C $	程式所選定的密碼
J , K	廻圈變數
B , W	此次猜測所得的黑數與白數
B $	操作者表現狀況的評語字串

研究建議

1. 修改程式內於遊戲結束後對操作者所做評語分析，加入密碼難易度及猜測次數的考慮。若是爲 4 位數每位數由 1 到 6 的密碼，則有 1296 種可能，但若爲 5 位數。每位數由 1 到 8 則有 32768 種可能。修改程式 2020 行到 2060行的訊息，使能依密碼的可能情形以及 G 值來訂出合理的評語訊息。

2. 遊戲開始時，讓操作者有機會自行決定密碼的難易度。可詢問操作者所希望的密碼位數，及各位數的數字範圍。必須確使輸入的爲有意義的數值——譬如，不可能產生零位數的密碼。另外一種方式爲詢問操作者想玩的困難等級爲簡單、中等、或困難。然後程式依此等級設定P值與D值。建議如下：

簡單（easy）：	D = 3 且 P = 3
中等（intermediate）：	D = 6 且 P = 4
困難（difficult）：	D = 8 且 P = 5

3. 評定操作者解碼猜數值的功力除了計算其猜測次數以外，尚可考慮他所花費的時間。欲加入此功能，必須修改程式190 行的 INPUT 。你必須於PEEK（－16384）後每次設A$字串內的一個字元來判斷是否有鍵按入。然後計算鍵未按入前廻圈執行的次數。依此即能對操作者的反應計時。

呻吟的骰子(GROAN)

可讓你呻吟的骰子戲

目 的

你是否喜歡緊張刺激的骰子遊戲？若你深好此道，則呻吟的骰子（ GROAN ）遊戲正合君意。此骰子戲爲兩人的遊戲，電腦爲你的對手與你勾心鬥角。此遊戲極大部分是靠運氣，然而如何當機立斷將你所擲骰子分數讓給你的對手，就必須靠一些智慧勇氣。

Apple 電腦在此爲雙方擲骰子，但不必擔心——它絕不會作弊（它是個正人君子，不會自我作賤的。）

你只要玩過一次後，就知道爲何要將此遊戲命名爲——呻吟的骰子。

使用方法

此遊戲使用兩顆骰子，它們類似市面上的六面骰子

，但有一點不同，骰子1點那一面爲一個縐眉的臉，其他五面則爲2點到6點。

遊戲的雙方先商量好目標分數，然後才進行遊戲，先到達目標分數的一方爲勝利者。雙方一回合一回合輪流玩，一回合依以下的規則擲骰子（至少擲一次，可能爲多次）。

只要兩個骰子都不出現縐眉臉，擲骰子的一方即具有繼續擲的權利。擲出一對無縐眉臉後，他可選擇再擲或者將擲骰子之權交給對手。若他將擲骰子權交給對手，他的原來分數即能加上此回合一系列擲骰子累積下來的分數。

但若他擲出一個“縐眉臉”，他就要縐眉了。若只出現一個“縐眉臉”，他就喪失此回合擲出的所有分數，但仍保留原來的分數，但若擲兩個“縐眉臉”，他就要縐眉加呻吟了，因爲他喪失了先前努力掙來的所有分數，重新回到零分，眞要失望透頂。

程式首先詢問雙方協調出的目標分數（在此處當然由操作者全權決定）。目標分數若選爲50到100 之間，可得最佳效果，但程式接受任何正整數。然後會出現一個模擬的丟銅板，隨和地決定由那一方開始擲骰子。

每次骰子擲出後，螢幕即顯示兩面骰子的圖形，表示擲出的結果。擲骰子之時，螢幕還會指出現在是那一方在擲。

擲完一次後，螢幕會顯示分數看板。此看板提供了所有相關資料：要贏得比賽所需的分數，此回合前雙方的分數，與此回合迄今擲出的累積分數。

若出現一個“縐眉臉”，看板即顯示此回合累積分數爲零。若出現兩個“縐眉臉”，則原來分數與此回合分數都爲零。在此兩種狀況下，擲骰子之權都會自動交給對方。

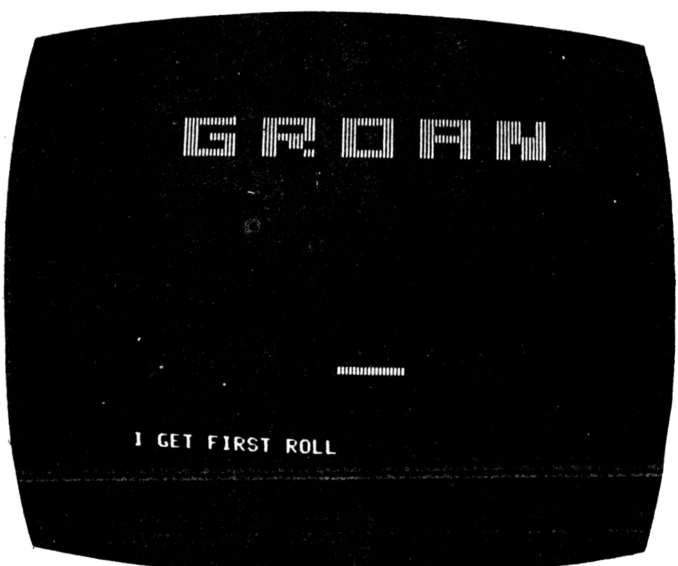
若擲出點數（即無“縐眉臉”），擲者必須決定是否再擲或讓出擲骰子權。程式內有一個秘密政策替Apple電腦一方決定如何判斷以贏得比賽。若你爲擲者，則於分數看板顯示後，會詢問你的決定。合法的回答爲P鍵與R鍵。R鍵表示你想要再擲，P鍵表示你讓出擲骰子權給電腦。若你想要累積分數以贏得比賽，有時就必須讓出擲骰子權以加上此回合的分數，免得偸雞不著蝕把米。

首先到達目標分數（可大於或等於）的一方爲勝利者，這也會讓他的對手惋惜呻吟一番。

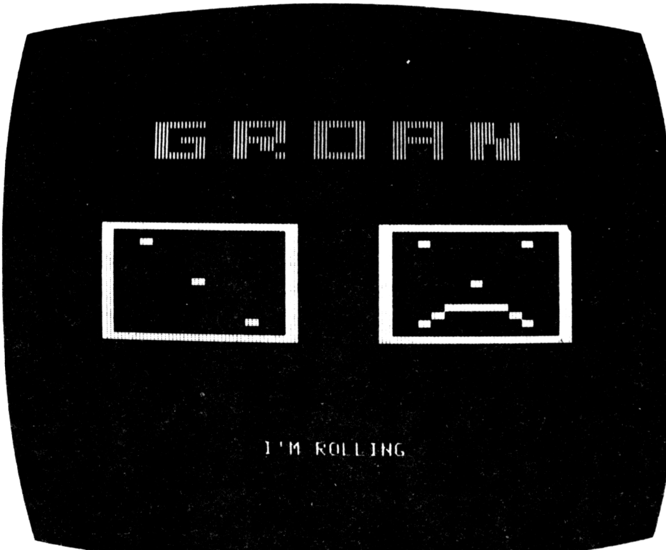
執行範例



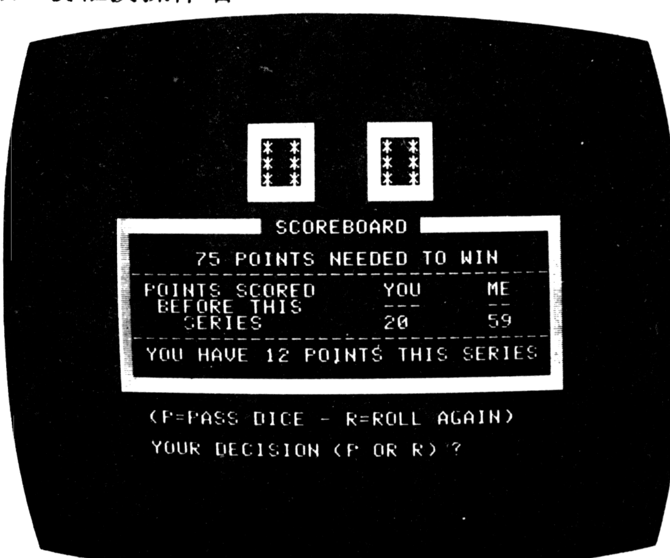
操作者決定挑戰Apple 75點的骰子戲。見彩色部分。



電腦贏得丟銅板。而取得先擲權。



電腦擲出骰子，得 3 點與“ 縐眉臉 ”。因此沒得到分數。擲權換操作者。



Much later in the same game, the operator rolls a 12 to start a series

遊戲進行不久後，操作者擲出12 點，開始繼續擲，此回合前的分數，Apple 爲59 點，操作者20 點。操作者必須決定是否再擲成移交擲權。

程 式

```
100 REM: GROAN
110 REM: COPYRIGHT 1980 BY PHIL FELDMAN AND TOM RUGG
150 A=0:H=0:T=0:B$=CHR$(32)
160 C$="*":BZ=-16336:Q=PEEK(78)+256*PEEK(79):Q=RND(-Q)
200 GR:HOME:GOSUB 2080:COLOR=C
210 XS=6:YS=2
220 GOSUB 2100:GOSUB 2200:DL=250:GOSUB 2050:GOSUB 2150
230 HOME:PRINT"HOW MUCH NEEDED TO WIN":PRINT
240 INPUT"(BETWEEN 50-100 IS BEST) ? ";W
250 W=INT(W):IF W<=0 THEN 230
260 GOSUB 2080:GOSUB 2500
270 IF Q>0.5 THEN 500
300 T=0
310 GOSUB 2700:T=T+R1+R2:IF F>0 THEN T=0
320 IF F=2 THEN H=0
330 GR:HOME:PRINT:PRINT TAB(13);"YOU'RE ROLLING"
340 P$="YOU":GOSUB 3200
350 IF F>0 THEN P$="ME":GOSUB 1560:DL=4000:GOSUB 2050:
    GOTO 500
360 GOSUB 1300:IF Q$="R" THEN 310
370 H=H+T:IF H>=W THEN 900
500 T=0
510 GR:HOME:GOSUB 2700
520 T=T+R1+R2:IF F>0 THEN T=0
530 IF F=2 THEN A=0
540 PRINT:PRINT TAB(15);"I'M ROLLING"
550 P$="I":GOSUB 3200
560 IF F>0 THEN P$="YOU":GOSUB 1560:DL=4000:GOSUB 2050:
    GOTO 300
570 HTAB XS:VTAB YS:GOSUB 5000
580 IF AD=1 THEN PRINT"I'LL ROLL AGAIN":DL=3500:GOSUB
    2050:GOTO 510
590 PRINT"I'LL STOP WITH THIS":A=A+T
600 IF A>=W THEN DL=3500:GOSUB 2050:GOTO 900
610 YS=YS+2:P$="YOU":GOSUB 1560:DL=3500:GOSUB 2050:GOTO
    300
900 XS=4:YS=2:HOME:P$="WE":T=0:GOSUB 1010
910 VTAB 16:HTAB 6:IF A>=W THEN PRINT
    "I WIN -- SKILL TRIUMPHS AGAIN"
920 IF H>=W THEN PRINT"YOU WIN -- IT WAS SHEER LUCK"
930 IF A>=W THEN GOSUB 2110
940 IF H>=W THEN BL=250:GOSUB 2000
950 VTAB 22:END
```

```
1000 TEXT:HOME:XS=4:YS=9
1010 VTAB YS:HTAB XS:INVERSE
1020 FOR J=1 TO 11:PRINT B$;:NEXT
1030 NORMAL:PRINT" SCOREBOARD ";:INVERSE
1040 FOR J=1 TO 11:PRINT B$;:NEXT
1050 FOR Y=YS TO YS+10:HTAB XS:VTAB Y:PRINT B$;
1060 HTAB XS+33:PRINT B$;:NEXT
1070 HTAB XS:FOR J=1 TO 34:PRINT B$;:NEXT
1080 NORMAL:HTAB XS+6:VTAB YS+2
1090 PRINT W;" POINTS NEEDED TO WIN";
1100 HTAB XS+1:VTAB YS+3:FOR J=1 TO 32
1110 PRINT"-";:NEXT
1120 HTAB XS+2:VTAB YS+4
1130 PRINT"POINTS SCORED      YOU     ME";
1140 HTAB XS+3:VTAB YS+5
1150 PRINT"BEFORE THIS        ---     --";
1160 HTAB XS+5:VTAB YS+6
1170 PRINT"SERIES";:HTAB XS+20:PRINT H;
1180 HTAB XS+28:PRINT A;
1190 HTAB XS+1:VTAB YS+7:FOR J=1 TO 32
1200 PRINT"-";:NEXT:HTAB XS+2:VTAB YS+8
1210 PRINT P$;" HAVE ";T;" POINTS THIS SERIES";
1220 NORMAL:RETURN
1300 HTAB XS:VTAB YS
1310 PRINT"(P=PASS DICE - R=ROLL AGAIN)"
1320 PRINT:HTAB XS
1330 PRINT"YOUR DECISION (P OR R) ?";:Q=POS(0)
1340 HTAB Q:GET Q$
1350 IF Q$="P" THEN RETURN
1360 IF Q$="R" THEN RETURN
1370 GOTO 1340
1400 INVERSE:FOR Y=YS-2 TO YS+2 STEP 4
1410 FOR X=XS-2 TO XS+2:GOSUB 1500:NEXT
1420 FOR X=XS+7 TO XS+11:GOSUB 1500:NEXT:NEXT
1430 FOR X=XS-2 TO XS+2 STEP 4
1440 FOR Y=YS-2 TO YS+2:GOSUB 1500:NEXT:NEXT
1450 FOR X=XS+19 TO XS+23 STEP 4:FOR Y=YS-6 TO YS-2
1460 FOR X=XS+7 TO XS+11 STEP 4:FOR Y=YS-2 TO YS+2
1470 GOSUB 1500:NEXT:NEXT:NORMAL:RETURN
1500 HTAB X:VTAB Y:PRINT B$;:RETURN
1520 HTAB X:VTAB Y:PRINT"G R O A N";:RETURN
1540 HTAB X:VTAB Y:PRINT"D E S P A I R !";:RETURN
1560 HTAB XS:VTAB YS:PRINT"DICE PASS TO ";P$:RETURN
1600 HTAB XS:VTAB YS:PRINT C$;:RETURN
1610 HTAB XS-1:VTAB YS-1:PRINT C$;
```

```
1620 HTAB XS+1:VTAB YS+1:PRINT C$;:RETURN
1630 GOSUB 1600:GOSUB 1610:RETURN
1640 HTAB XS+1:VTAB YS-1:PRINT C$;
1650 HTAB XS-1:VTAB YS+1:PRINT C$;:GOSUB 1610:RETURN
1660 GOSUB 1600:GOSUB 1640:RETURN
1670 HTAB XS-1:VTAB YS:PRINT C$;:HTAB XS+1
1680 PRINT C$;:GOSUB 1640:RETURN
1700 HTAB XS-1:VTAB YS-1:PRINT"@ @";
1710 HTAB XS:VTAB YS:PRINT"↑";
1720 HTAB XS-1:VTAB YS+1
1730 PRINT"///";:RETURN
1800 LX=XS-L:RX=XS+L:UY=YS-L:LY=YS+L
1810 HLIN LX,RX AT UY:HLIN LX,RX AT LY
1820 VLIN UY,LY AT LX:VLIN UY,LY AT RX:RETURN
1900 PLOT XS,YS:RETURN
1910 PLOT XS-4,YS-5:PLOT XS+4,YS+5:RETURN
1920 GOSUB 1900:GOSUB 1910:RETURN
1930 PLOT XS-4,YS+5:PLOT XS+4,YS-5:GOSUB 1910:RETURN
1940 GOSUB 1900:GOSUB 1930:RETURN
1950 PLOT XS-4,YS:PLOT XS+4,YS:GOSUB 1930:RETURN
1960 GOSUB 1940:FOR J=1 TO 3
1970 PLOT XS-J-1,YS+J+2:PLOT XS+J+1,YS+J+2
1980 NEXT:PLOT XS-1,YS+3:PLOT XS,YS+3:PLOT XS+1,YS+3
1990 RETURN
2000 FOR J=1 TO BL:Q=PEEK(BZ):NEXT:RETURN
2050 FOR J=1 TO DL:NEXT:RETURN
2080 C=RND(1)*14+2:RETURN
2100 BL=120:GOSUB 2000:DL=100:GOSUB 2050
2110 BL=40:DL=40
2120 FOR K=1 TO 2:GOSUB 2050:GOSUB 2000
2130 NEXT:DL=60:BL=70:GOSUB 2050:GOSUB 2000
2140 RETURN
2150 BL=200:GOSUB 2000:RETURN
2200 HLIN XS,XS+3 AT YS:VLIN YS,YS+4 AT XS
2210 HLIN XS,XS+3 AT YS+4:VLIN YS+2,YS+4 AT XS+3
2220 PLOT XS+2,YS+2:VLIN YS,YS+4 AT XS+6
2230 HLIN XS+6,XS+9 AT YS:HLIN XS+6,XS+9 AT YS+2
2240 PLOT XS+9,YS+1:PLOT XS+8,YS+3
2250 PLOT XS+9,YS+4:HLIN XS+12,XS+15 AT YS
2260 HLIN XS+12,XS+15 AT YS+4:VLIN YS,YS+4 AT XS+12
2270 VLIN YS,YS+4 AT XS+15:VLIN YS,YS+4 AT XS+18
2280 VLIN YS,YS+4 AT XS+21:HLIN XS+18,XS+21 AT YS
2290 HLIN XS+18,XS+21 AT YS+2:VLIN YS,YS+4 AT XS+24
2300 VLIN YS,YS+4 AT XS+27:VLIN YS,YS+2 AT XS+25
2310 VLIN YS+2,YS+4 AT XS+26:RETURN
```

```
2500 DL=25:X=20:Y=33
2510 HOME:PRINT"AND NOW A COIN TOSS FOR FIRST ROLL"
2520 PRINT:PRINT"THE COIN IS IN THE AIR AND . . . ."
2530 Y=Y-1:COLOR=C:GOSUB 2650:GOSUB 2050
2540 COLOR=0:GOSUB 2650:COLOR=C:GOSUB 2660:GOSUB 2050
2550 COLOR=0:GOSUB 2660:IF Y>9 THEN 2530
2560 Y=Y+1:COLOR=C:GOSUB 2650:GOSUB 2050
2570 COLOR=0:GOSUB 2650:COLOR=C:GOSUB 2660:GOSUB 2050
2580 COLOR=0:GOSUB 2660:IF Y<33 THEN 2560
2590 COLOR=C:Y=33:GOSUB 2650:DL=50:GOSUB 2000
2600 Q=RND(1):Q$="YOU":IF Q>.5 THEN Q$="I"
2610 HOME:PRINT
2620 PRINT B$;B$;Q$;B$;"GET FIRST ROLL"
2630 DL=3000:GOSUB 2050:RETURN
2650 HLIN X-2,X+2 AT Y:RETURN
2660 VLIN Y-2,Y+2 AT X:RETURN
2700 R1=INT(RND(1)*6+1):R2=INT(RND(1)*6+1)
2710 F=0:IF R1=1 THEN F=1
2720 IF R2=1 THEN F=F+1
2730 RETURN
2800 FOR J=0 TO 14:L=J/2
2810 XS=9:YS=8+J:COLOR=C:GOSUB 1800
2820 XS=30:GOSUB 1800:COLOR=0:XS=9:GOSUB 1800
2830 XS=30:GOSUB 1800:NEXT
2840 COLOR=C:XS=9:GOSUB 1800
2850 XS=30:GOSUB 1800:RETURN
3000 XS=16:YS=5:GOSUB 1400:Q=R1:GOSUB 3100
3010 XS=25:Q=R2:GOSUB 3100
3020 IF R1=1 THEN X=4:Y=YS:GOSUB 1520
3030 IF R2=1 THEN X=29:Y=YS:GOSUB 1520
3040 IF F>0 THEN GOSUB 2150
3050 IF F<2 THEN RETURN
3060 DL=400:GOSUB 2050:GOSUB 2100:DL=800:GOSUB 2050
3070 X=14:Y=1:GOSUB 1540:GOSUB 2150:RETURN
3100 ON Q GOSUB 1700,1610,1630,1640,1660,1670:RETURN
3200 GOSUB 2080:GOSUB 2800:XS=9:YS=22:Q=R1:GOSUB 3250
3210 XS=30:Q=R2:GOSUB 3250:GOSUB 2080:COLOR=C:XS=6
3220 YS=2:IF F>0 THEN GOSUB 2200:GOSUB 2150:GOTO 3240
3230 DL=1000:GOSUB 2050
3240 GOSUB 1000:GOSUB 3000:XS=6:YS=21:RETURN
3250 ON Q GOSUB 1960,1910,1920,1930,1940,1950:RETURN
5000 V=A+T:IF V>=W THEN 5100
5010 IF (W-H)<10 THEN 5110
5020 IF A>=H THEN CT=T/25:GOTO 5050
5030 IF V<H THEN CT=T/35:GOTO 5050
```

```
5040 CT=T/30
5050 IF RND(1)>CT THEN 5110
5100 AD=0:RETURN
5110 AD=1:RETURN
```

修改程式

1. 若你想固定遊戲的目標分數，可修改230行以及刪除240行。只要設W的值爲目標分數即可。例如：

```
230 W=100: HOME
```

即可設目標分數爲100 。不要忘了刪除240行。

2. 若想刪除每次擲骰子之前所顯示的圖形，可直接刪除2810行， 2820行，2830行， 2840 行，以及修改2800行和2850行如下：

```
2800 YS=22:L=7:COLOR=C
2850 XS=9:GOSUB 1800:XS=30:GOSUB 1800:RETURN
```

如此可加快遊戲的進行速度。

3. 此遊戲當你玩幾次以後，也許想要加遊戲的進行，可修改顯示各個訊息時的時間停頓。欲使遊戲加速，可試下面：

```
2050 DL=DL/2:FOR J=1 TO DL:NEXT
2055 DL=2*DL:RETURN
```

欲使遊戲減慢，可試下面：

```
2050 DL=2*DL:FOR J=1 TO DL:NEXT
2055 DL=DL/2:RETURN
```

4. 在此程式中，顯像的顏色爲隨機選擇。若你想用特定的顏色，譬如：橘色一做爲你的骰子；紫色一做爲Apple 電腦的骰子，可做以下的改變。

```
2080 C=RND(1)*14+2
2082 IF P$="YOU" THEN C=9
2084 IF P$="I" THEN C=11
2086 RETURN
```

當然，你可任選，Applesoft 手册所描述的15種顏色，只要使用COLOR =指令即可。選擇骰子的顏色在2082行，而電腦的骰子顏色的選擇在2084行。

5. 目前寫成的程式無法容納於16K 使用卡式錄音帶的Applesoft 電腦（見附錄1）。然而若套用"修改程式"第二題所建議的方法，即能將程式濃縮，放入此類系統。亦即必須删除2210行到2310行；删除2540行到2590行，以及加入：

```
2200 RETURN
2500 REM
2530 DL=4000:GOSUB 2050
```

這些改變犧牲了此程式某些迷人有趣的部分，但實質上並沒改變程式的性能。

主流程

150-160	設定常數
200-270	顯示遊戲開始，取得目標分數
300-370	操作者擲骰子
500-610	電腦擲骰子
900-950	結束訊息
1000-1220	顯示分數看板
1300-1370	詢問操作者的重擲決定
1400-1470	畫出看板及骰子的邊綫
1500-1560	副程式，印出各種訊息
1600-1730	畫出看板骰子的面
1800-1820	畫出骰子圖形的邊綫
1900-1990	畫出骰子圖形的面
2000-2150	停頓廻圈及嗡嗡聲控制廻圈
2200-2310	畫遊戲標題" GROAN "
2500-2660	丟銅板
2700-2730	判斷所擲的骰子
2800-2850	控制骰子圖形
3000-3100	控制看板的顯像
3200-3250	副程式，控制看板及骰子面
5000-5110	Apple 電腦的秘密決策，設AD = 0 以停止擲骰子。或AD = 1 以繼續擲骰子

主要變數

A	Apple 電腦前幾回合的總分
H	操作者前幾回合的總分
T	目前回合的分數
B$	一個空白字元的字串
C$	一個星號字元的字串
BZ	嗡嗡聲產生器的引數
C	圖形的顏色
XS,YS	螢幕水平及垂直的印出參考位置
W	目標分數
Q	工作變數
R1,R2	骰子1及骰子2擲出的點數
F	擲出的結果（0＝無“綴眉臉”，1＝一個“綴眉臉，2＝兩個“綴眉臉”）
DL	停頓的長度
BL	嗡嗡叫的長度
P$	目前擲者的名字字串
Q$	工作字串變數
AD	電腦的決策標示（0＝停止擲，1＝繼續擲）
J,K	迴圈指標
X,Y	螢幕的水平及垂直印出位置
L	骰子一邊長度的一半
LX,XY	一個骰子面的上邊與下邊位置

V	若讓出擲骰子權，Apple 電腦所能獲得的分數
CT	在Apple 電腦的秘密決策中，所用的臨界值

研究建議

1. 電腦的秘密決策位於程式5000 行以後。每當電腦擲出無"豬眉臉"後，它必須決定是否繼續擲。希望你能進一步改進目前的決策方式。你可使用，但不能修改，變數A，T，H，與W。而AD 變數必須於歸回前設定好。設AD = 0 表示Apple 讓出骰子；或AD = 1 表示Apple 要再擲骰子。
2. 詢問操作者的姓名，然後在看板或其它訊息中印出其稱呼。
3. 請探討與擲骰子連結一起的圖形程序工作情形，以便改進它們，使能更具眞實臨場感。

猜字(JOT)

向電腦挑戰猜字

目　的

JOT　猜字遊戲爲兩人對抗的遊戲，其牽涉到不少心智的推導。Apple 電腦在此猜字遊戲中爲你的對手，請勿輕敵！你會發現你的對手爲相當厲害的角色。

猜字遊戲的規則相當簡單，此程式一律玩三個字母的單字，且單字內的字母不能重覆（可見“修改程式”內所提到的另一種合法字的標準）。

遊戲開始前，遊戲雙方都選一個秘密字。接著就開始猜對方的秘密字，先猜出者獲勝。

遊戲雙方輪流猜測其對方的秘密字。在每次猜測後，對方必須告訴問者他猜對的字母個數，字母在單字中的位置則沒關緊要。例如：若秘密字爲“ own ”而猜“ who ”一字，則爲猜中兩個字母，不管位置。

使用方法

程式首先顯示遊戲簡介，然後要求你想好自己的秘密字。然後問你是否要開始猜第一次。你猜完後，就換電腦猜你的秘密字。

電腦猜一字後，即詢問猜測的結果，你可按入 0 ， 1 ， 2 ， 3 ，與R 鍵。R 鍵表示Apple 電腦已猜中你的秘密字——相當令人臉紅的一件事。 0 到 3 這些數字的回答表示電腦所猜中的字母個數（不管位置）。若回答了，即爲三個字母都猜對，但必須將其字母的順序重排方能與秘密字相同。譬如，猜“ EAT ” 而秘密字爲“ TEA ”。

取得這些資料以後，電腦必須花一些時間來處理。若時間不夠，它會顯示訊息“ I'M THINKING”，因此你不要誤認爲它又在發呆做白日夢。若電腦發現有不一致的現象時，它會詢問你的秘密字，找出錯誤之處。

當輪到你猜時，你可使用兩種回答，一爲S 鍵，一爲Q 鍵。S 鍵爲總結，整理出你前幾次所猜的寫以及其對應的命中情形。當你要分析前幾次猜測所累積的資料時，此相當有用。S 鍵按完後，電腦會再摧促你趕快做猜字。Q 鍵爲放棄，即結束遊戲。

若不使用這兩個特別鍵，你可輸入猜測的字。當然，此猜測字必定爲三個字母的單字，若你按入的字非法，電腦會提醒你。做一個合法的猜測後，電腦會告訴你命中多少個字母，若你猜到電腦的秘密字，電腦就會打

出恭維賀喜的訊息。然後Apple 電腦會詢問你的秘密字，檢查它的錯誤及你給的命中資料是否正確。

執行範例

```
            JOT

JUST A MOMENT PLEASE.....

THANKS, NOW LET'S EACH THINK
OF OUR SECRET WORD

(THIS TAKES ME A WHILE...)

I'VE ALMOST GOT IT...
```

```
OK, DO YOU WANT TO GO FIRST? NO

MY GUESS IS -- NIP
HOW DID I DO (O-3 OR R)? 1

I'M THINKING...

YOUR GUESS (OR S OR Q)? DOG
# OF HITS IS 1

MY GUESS IS -- NOR
HOW DID I DO (O-3 OR R)? O

I'M THINKING...
              .
              .
              .
      (later in the same game)
              .
              .
              .
YOUR GUESS (OR S OR Q)? S

-------------------------------------
YOUR GUESSES    SUMMARY    MY GUESSES
WORD   HITS                WORD  HITS
 DOG    1          1        NIP   1
 CAT    0          2        NOR   0
 LIP    0          3        DIG   0
 SON    0          4        PUT   2
                   5        PUB   1

YOUR GUESS (OR S OR Q)? FED
# OF HITS IS 2

MY GUESS IS -- PET
HOW DID I DO (O-3 OR R)? R

IT SURE FEELS GOOD

MY WORD WAS - WED

HOW ABOUT ANOTHER GAME? NO
```

程 式

```
100 REM: JOT
110 REM: COPYRIGHT 1980 BY PHIL FELDMAN AND TOM RUGG
150 M=25:N=406
160 DIM A$(N)
170 DIM G1$(M),G2$(M),H1(M),H2(M)
200 G1=0:G2=0
210 L=N:Q=PEEK(78)+256*PEEK(79):Q=RND(-Q)
250 TEXT:HOME:PRINT TAB(17);"J O T":PRINT
260 PRINT"JUST A MOMENT PLEASE .....":GOSUB 3000:Q=RND(
    1)*N+1:PRINT
270 PRINT"THANKS, NOW LET'S EACH THINK":PRINT
    "OF OUR SECRET WORD"
280 PRINT:PRINT"(THIS TAKES ME A WHILE ...)"
290 GOSUB 2200:M$=A$(Q):PRINT:PRINT"OK, ";
300 INPUT"DO YOU WANT TO GO FIRST? ";Q$
310 Q$=LEFT$(Q$,1):IF Q$="N" THEN 600
320 IF Q$="Y" THEN 500
330 PRINT:PRINT"YES OR NO PLEASE":PRINT:GOTO 300
500 PRINT:INPUT"YOUR GUESS (OR S OR Q)? ";P$:IF P$="S"
     THEN GOSUB 1000:GOTO 500
510 IF P$="Q" THEN 1200
520 IF P$=M$ THEN G1=G1+1:G1$(G1)=P$:H1(G1)=9:GOTO 3400
530 GOSUB 1800:IF F=0 THEN PRINT
    "THAT'S NOT A LEGAL WORD -- TRY AGAIN":GOTO 500
540 Q$=M$:GOSUB 2600:Q$=P$:GOSUB 1500
550 PRINT"# OF HITS IS ";Q
560 G1=G1+1:G1$(G1)=Q$:H1(G1)=Q
570 IF G1=M THEN 3600
600 Q$=A$(L):G2=G2+1:G2$(G2)=Q$
610 PRINT:PRINT"MY GUESS IS -- ";Q$
620 INPUT"HOW DID I DO (0-3 OR R)? ";P$
630 P$=LEFT$(P$,1)
640 IF P$="R" THEN H2(G2)=9:GOTO 3200
650 P=VAL(P$):IF P>3 OR (P=0 AND P$<>"0") THEN PRINT
    "BAD ANSWER":GOTO 610
660 IF L>100 THEN PRINT:PRINT"I'M THINKING ..."
670 H2(G2)=P:GOSUB 800
680 GOTO 500
800 Q$=G2$(G2):H=H2(G2):J=0:GOSUB 2600:L=L-1:IF L<1
     THEN 900
810 J=J+1:IF J>L THEN 870
820 Q$=A$(J):GOSUB 1500
830 IF Q=H THEN 810
```

```
840 A=J:B=L:GOSUB 2400:L=L-1
850 IF L<1 THEN 900
860 IF L>=J THEN 820
870 RETURN
900 PRINT:PRINT"SOMETHING'S WRONG !!"
910 PRINT:INPUT"WHAT'S YOUR SECRET WORD? ";P$:GOSUB
    1800
920 IF F=0 THEN PRINT:PRINT
    "ILLEGAL WORD -- I NEVER HAD A CHANCE":GOTO 1200
930 PRINT:PRINT"YOU GAVE A BAD ANSWER SOMEWHERE --"
940 PRINT"CHECK THE SUMMARY":GOSUB 1000
950 GOTO 1200
1000 PRINT:Q=G1:IF G2>G1 THEN Q=G2
1010 IF Q=0 THEN PRINT"NO GUESSES YET":RETURN
1020 PRINT TAB(1);:FOR J=1 TO 37:PRINT"-";:NEXT:PRINT"-"
1030 PRINT TAB(1);:INVERSE:PRINT"YOUR GUESSES";:NORMAL
1040 PRINT"    SUMMARY     ";:INVERSE
1050 PRINT"MY GUESSES":NORMAL:PRINT TAB(1);
1060 PRINT"WORD   HITS";TAB(29);"WORD  HITS"
1070 FOR J=1 TO Q:K=1:IF J>9 THEN K=0
1080 IF J>G1 THEN PRINT TAB(19+K);J;TAB(30);G2$(J);
     TAB(36);H2(J):GOTO 1110
1090 IF J>G2 THEN PRINT TAB(2);G1$(J);TAB(9);H1(J);
     TAB(19+K);J:GOTO 1110
1100 PRINT TAB(2);G1$(J);TAB(9);H1(J);TAB(19+K);J;
     TAB(30);G2$(J);TAB(36);H2(J)
1110 NEXT:RETURN
1200 PRINT:INPUT"HOW ABOUT ANOTHER GAME? ";Q$
1210 Q$=LEFT$(Q$,1):IF Q$="Y" THEN 200
1220 IF Q$="N" THEN END
1230 PRINT:PRINT"YES OR NO PLEASE":GOTO 1200
1500 P$=LEFT$(Q$,1):Q=0:GOSUB 1600
1510 P$=MID$(Q$,2,1):GOSUB 1600
1520 P$=RIGHT$(Q$,1):GOSUB 1600:RETURN
1600 IF P$=M1$ OR P$=M2$ OR P$=M3$ THEN Q=Q+1
1610 RETURN
1800 F=0
1810 FOR J=1 TO N
1820 IF A$(J)=P$ THEN F=1:RETURN
1830 NEXT:RETURN
2200 FOR A=N TO 100 STEP -1:B=INT(RND(1)*A)+1
2210 GOSUB 2400:NEXT
2220 PRINT:PRINT"I'VE ALMOST GOT IT ..."
2230 FOR A=99 TO 2 STEP -1:B=INT(RND(1)*A)+1
2240 GOSUB 2400:NEXT:RETURN
```

```
2400 Q$=A$(B):A$(B)=A$(A):A$(A)=Q$:RETURN
2600 M1$=LEFT$(Q$,1):M2$=MID$(Q$,2,1)
2610 M3$=RIGHT$(Q$,1):RETURN
3000 RESTORE:FOR P=1 TO N:READ A$(P):NEXT:RETURN
3200 PRINT:PRINT"IT SURE FEELS GOOD"
3210 PRINT:PRINT"MY WORD WAS -- ";M$
3220 GOTO 1200
3400 PRINT:PRINT"CONGRATULATIONS - THAT WAS IT":PRINT
3410 INPUT"WHAT WAS YOUR WORD? ";P$:GOSUB 1800:J=1
3420 IF F=0 THEN PRINT:PRINT
     "ILLEGAL WORD - I HAD NO CHANCE":GOTO 1200
3430 IF A$(J)=P$ THEN PRINT:PRINT"NICE WORD":GOTO 1200
3440 J=J+1:IF J<=L THEN 3430
3450 PRINT:PRINT"YOU MADE AN ERROR SOMEWHERE":PRINT
     "-- CHECK THE SUMMARY"
3460 GOSUB 1000:GOTO 1200
3600 PRINT:PRINT"SORRY, I M OUT OF MEMORY":PRINT
3610 PRINT"MY WORD WAS - ";M$:GOTO 1200
5000 DATA  ACE,ACT,ADE,ADO,ADS,AFT,AGE
5010 DATA  AGO,AID,AIL,AIM,AIR,ALE,ALP
5020 DATA  AND,ANT,ANY,APE,APT,ARC,ARE
5030 DATA  ARK,ARM,ART,ASH,ASK,ASP,ATE
5040 DATA  AWE,AWL,AXE,AYE,BAD,BAG,BAN
5050 DATA  BAR,BAT,BAY,BED,BEG,BET,BID
5060 DATA  BIG,BIN,BIT,BOA,BOG,BOW,BOX
5070 DATA  BOY,BUD,BUG,BUM,BUN,BUS,BUT
5080 DATA  BUY,BYE,CAB,CAD,CAM,CAN,CAP
5090 DATA  CAR,CAT,COB,COD,COG,CON,COP
5100 DATA  COT,COW,COY,CRY,CUB,CUD,CUE
5110 DATA  CUP,CUR,CUT,DAB,DAM,DAY,DEN
5120 DATA  DEW,DIE,DIG,DIM,DIN,DIP,DOE
5130 DATA  DOG,DON,DOT,DRY,DUB,DUE,DUG
5140 DATA  DYE,DUO,EAR,EAT,EGO,ELK,ELM
5150 DATA  END,ELF,ERA,FAD,FAG,FAN,FAR
5160 DATA  FAT,FED,FEW,FIG,FIN,FIR,FIT
5170 DATA  FIX,FLY,FOE,FOG,FOR,FOX,FRY
5180 DATA  FUN,FUR,GAP,GAS,GAY,GEM,GET
5190 DATA  GIN,GNU,GOB,GOD,GOT,GUM,GUN
5200 DATA  GUT,GUY,GYP,HAD,HAG,HAM,HAS
5210 DATA  HAT,HAY,HEN,HEX,HID,HIM,HIP
5220 DATA  HIS,HIT,HER,HEM,HOE,HOG,HOP
5230 DATA  HOT,HOW,HUB,HUE,HUG,HUM,HUT
5240 DATA  ICE,ICY,ILK,INK,IMP,ION,IRE
5250 DATA  IRK,ITS,IVY,JAB,JAR,JAW,JAY
5260 DATA  JOB,JOG,JOT,JOY,JUG,JAG,JAM
```

```
5270 DATA  JET,JIB,JIG,JUT,KEG,KEY,KID
5280 DATA  KIN,KIT,LAB,LAD,LAG,LAP,LAW
5290 DATA  LAY,LAX,LED,LEG,LET,LID,LIE
5300 DATA  LIP,LIT,LOB,LOG,LOP,LOT,LOW
5310 DATA  LYE,MAD,MAN,MAP,MAR,MAT,MAY
5320 DATA  MEN,MET,MID,MOB,MOP,MOW,MUD
5330 DATA  MIX,MUG,NAB,NAG,NAP,NAY,NET
5340 DATA  NEW,NIL,NIP,NOD,NOT,NOR,NOW
5350 DATA  NUT,OAF,OAK,OAR,OAT,ODE,OIL
5360 DATA  OLD,ONE,OPT,ORE,OUR,OUT,OVA
5370 DATA  OWE,OWL,OWN,PAD,PAL,PAN,PAR
5380 DATA  PAT,PAW,PAY,PEA,PEG,PEN,PET
5390 DATA  PEW,PIE,PIG,PIT,PLY,POD,POT
5400 DATA  POX,PER,PIN,PRO,PRY,PUB,PUN
5410 DATA  PUS,PUT,RAG,RAM,RAN,RAP,RAT
5420 DATA  RAW,RAY,RED,RIB,RID,REV,RIG
5430 DATA  RIM,RIP,ROB,ROD,ROE,ROT,ROW
5440 DATA  RUB,RUE,RUG,RUM,RUN,RUT,RYE
5450 DATA  SAD,SAG,SAP,SAT,SAW,SAY,SET
5460 DATA  SEW,SEX,SHY,SEA,SIN,SHE,SIP
5470 DATA  SIR,SIT,SIX,SKI,SKY,SLY,SOB
5480 DATA  SOD,SON,SOW,SOY,SPA,SPY,STY
5490 DATA  SUE,SUM,SUN,TAB,TAD,TAG,TAN
5500 DATA  TAP,TAX,TAR,TEA,TEN,THE,THY
5510 DATA  TIC,TIE,TIN,TIP,TOE,TON,TOP
5520 DATA  TOW,TOY,TRY,TUB,TUG,TWO,URN
5530 DATA  USE,UPS,VAN,VAT,VEX,VIA,VIE
5540 DATA  VIM,VOW,YAK,YAM,YEN,YES,YET
5550 DATA  YOU,WAD,WAG,WAN,WAR,WAS,WAX
5560 DATA  WAY,WEB,WED,WET,WHO,WHY,WIG
5570 DATA  WIN,WIT,WOE,WON,WRY,ZIP,FIB
```

修改程式

1. 通常操作者在做下一個猜測時，都要先看看前幾回猜字的總結資料。因此若你要程式在操作者猜測前，自動顯示總結資料，可修改 500 行爲：

```
500 GOSUB 1000:PRINT:INPUT
          "YOUR GUESS (OR Q)?";P$
```

2. 你可更改猜測次數的上限，只要改變150行的M值即可。你也許想增加猜測次數的上限，以便和“研究計劃”第2題相配合。你也許想減少猜測次數的上限，以節省記錄空間。目前M的值為 25 ，比實際所需稍大。遊戲實際進行很少超過15 次猜測。若欲設M為15 ，可更改150行為：

```
150 M=15:N=406
```

3. 可很輕易地修改合法字的資料列表。我們對合法字的要求標準如下：它們必須為三個不同的字母，而且不能為：

—以大寫字母為首
—簡寫
—感嘆詞(譬如“ ugh ”，“ hey ”等等)
—專業名詞(譬如“ ohm ”，“ sac ”，“ yaw ”等等)

在150行，N設為資料列表的字數總和。資料列表本身是由5000 行開始。

欲加入新字，可依以下的步驟：將新的單字置於目前資料列表尾部的DATA敍述內(使用大於5570的行號)。然後設定N的值為406加上新加入的單字數目。例如，欲加“ ohm ”與“ yaw ”於列表內

，可修改 150 行爲：

150 M=25:N=408

然後加入新行

5580 DATA OHM,YAW

欲刪除單字，可依相反方式完成。搬掉DATA敍述內的單字，然後依此減少N 的值。

主流程

150-170	設陣列的維數
200-330	遊戲開始
500-570	操作者猜電腦的秘密字
600-680	電腦猜操作者的秘密字
800-870	推演操作者可能的秘密字，將它們移到A$陣列的先頭
900-950	處理所獲得資料的不一致現象
1000-1110	顯示目前的資料總結列表
1200-1230	詢問再玩一次
1500-1610	受猜的字與秘密字比較
1800-1830	檢查是否輸入的字爲合法
2200-2240	將A$陣列的內容隨機打散
2400	互換A$陣列的元素A與元素B
2600-2610	將Q$字串切割成各個字母
3000	由DATA敍述讀入A$陣列

3200-3220	Apple 電腦贏後的善後工作
3400-3460	操作者勝利後的善後工作
3600-3610	錯誤程序——猜的次數超過上限
5000-5570	資料

主要變數

N	資料單字的總數
M	猜測次數的上限
A$	存放資料單字的字串陣列
G1$,G2$	操作者與電腦猜測字的字串陣列
H1,H2	操作者與電腦對應到G1$與G2$的命中情形陣列
G1,G2	操作者與電腦目前猜測的次數
M$	電腦的秘密字
M1$,M2$,M3$	單字的第一，第二，及第三字母
P$,Q$	工作字串變數
L	目前操作者的可能秘密字的個數
F	輸入單字是否合法的標示
H	最近一次猜測的命中字母個數
A,B	A$陣列內欲互換內容的位置
J,P,Q	工作變數，陣列與迴圈的註標
K	顯示總結列表的格式變數

研究建議

1. 可爲程式添加一些有趣的訊息，使遊戲的進行更生動有趣。例如，當Apple 電腦慢慢縮小你的秘密字的範圍，而快要猜出你的秘密字時，可偶而打出下面訊息：

   JUST AS I THOUGHT . . .
   HMM, I DIDN'T EXPECT THAT . . .
   JUST WHAT I WAS HOPING TO HEAR . . .

   L 變數的值即爲電腦縮小你的可能秘密字範圍的值。你可定期地檢查L 值，當其值很小時，可打出下面的訊息：

   BE CAREFUL, I'M CLOSING IN ON YOU.

2. 加入一個功能，使輸的那一方仍能繼續猜對方的字。不論G1 與G2的值相差極多，總結顯像程序仍能正常的工作。且若猜中秘密字，則命中字母值印出9。
3. 使單字內字母的重覆變爲合法，亦即使秘密字與猜測字都能有字母的重覆。此牽涉到修改單字列表，加入新的單字，修改程式使能允許此類的猜測字輸入。

障礙賽(OBSTACLE)

跟你的朋友一起玩障礙賽跑

目　的

此程式可讓你和你的朋友（或對手）一起玩障礙賽的遊戲（OBSTACLE）。此遊戲綜合了反應技術（即反射速度，手到眼之間的配合等等）以及遊戲技巧，方能擊敗你的助手。每次遊戲通常只花一兩分鐘就分出勝負，因此你必須先有充分的心理準備，免得到時候手忙脚亂。

使用方法

此遊戲以最先撞到障礙物的那一方爲輸方，因此誰撐得越久，不撞到障礙，誰就勝利。程式開始後，即詢問左右雙方的名字。然後顯示競賽場，以及雙方的起跑點，接著你按任何一鍵即開始比賽，按ESCAPE 鍵即

結束比賽。

按任何一鍵開始比賽後，雙方以任意四個方向——上，下，左，右，開始前進。當玩者前進時，即隨而在他走過的軌跡建立一道牆。由電腦決定雙方前進的速度，玩者只能控制他自己的方向。左方的玩者可按W，Z，A或S鍵分別控制上，下，左，右四個方向。右方的玩者可按P，•（句點），L與；（分號）鍵分別控制上，下，左，右。你可很容易地在Apple 電腦的鍵盤上發現這些鍵，練習一下即能習慣各鍵的功能。

首先碰到競賽場的邊牆以及雙方築好的牆之玩者爲輸家。若發生撞牆時，程式會表明撞擊失事的那一點，然後顯示勝利一方的名字。然後遊戲從頭開始。

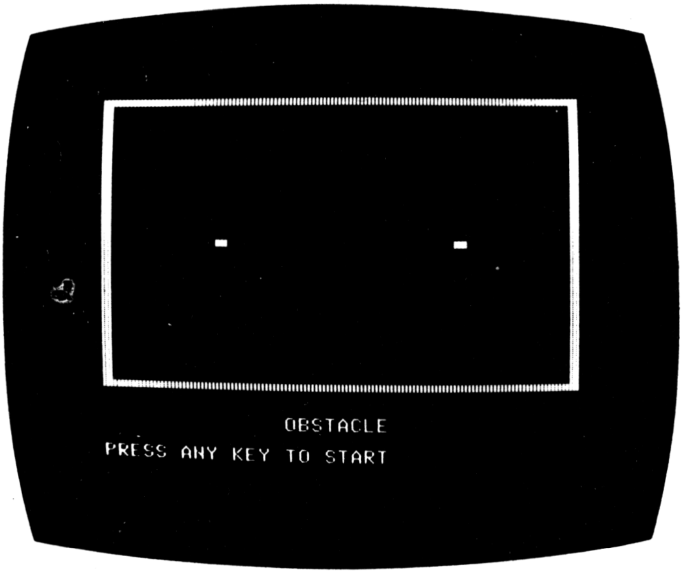
要玩好此遊戲的原則是相當有趣的。若你想攻擊你的對手，就想辦法用牆將他圍起來，他就自然在其中撞牆失事。或者你可以跟他離遠一點，然後充分利用你所佔據的空間，直到你的對手耗盡他自己的空間。可分別嘗試這兩個方法看看何者較優。

當按一鍵欲改變方向時，應注意快按一下後即可放掉，不要連續按住——因也許你會干擾電腦接受對方的鍵。因爲通常電腦一次只能接收一個鍵的訊號。

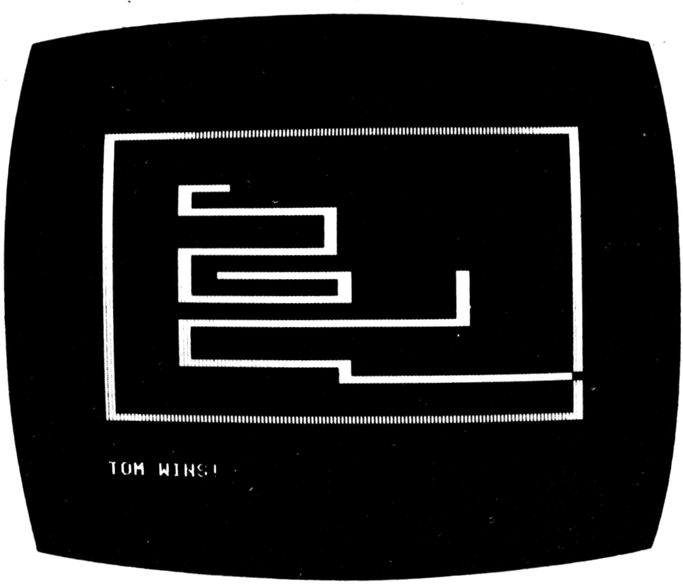
執行範例



程式開始，詢問遊戲雙方的名字



程式畫出競賽場，等待按鍵以開始遊戲。見彩色部分。



遊戲開始，雙方往隨機方向行走，右方的 Phil 沒改變方向以致撞牆。Tom 贏。

程 式

```
100 REM: OBSTACLE
110 REM: COPYRIGHT 1980 BY TOM RUGG AND PHIL FELDMAN
120 GOSUB 600
125 GR
130 PRINT TAB(15);"OBSTACLE"
140 PRINT
150 PRINT"PRESS ANY KEY TO START"
155 Z=15:F=-16384:S=-16368:T=36
160 AX=INT(T/4):AY=INT(T/2):BX=3*AX:BY=AY:A=7:B=13
165 E=12:AD=INT(4*RND(1))+1:BD=INT(4*RND(1))+1
170 GOSUB 950:GOSUB 900
175 C=PEEK(F):POKE S,0
180 C=PEEK(F):IF C<128 THEN 180
190 POKE S,0:HOME
195 IF C=155 THEN END
200 GOSUB 950:GOSUB 900
205 FOR J=1 TO 5:C=PEEK(F):POKE S,0:NEXT J
210 X=AX:Y=AY:D=AD:GOSUB 1000
220 AR=R:AX=X:AY=Y
230 X=BX:Y=BY:D=BD:GOSUB 1000
240 BR=R:BX=X:BY=Y
245 IF AR=1 OR BR=1 THEN 400
250 GOSUB 900
260 FOR J=1 TO 6:C=PEEK(F):POKE S,0
270 IF C=215 THEN AD=1
280 IF C=218 THEN AD=2
290 IF C=193 THEN AD=3
300 IF C=211 THEN AD=4
310 IF C=208 THEN BD=1
320 IF C=174 THEN BD=2
330 IF C=204 THEN BD=3
340 IF C=187 THEN BD=4
350 NEXT J:GOTO 210
400 GOSUB 700:X=AX:Y=AY
410 IF AR=1 THEN 420
415 X=BX:Y=BY
420 FOR J=1 TO 15
430 COLOR=Z:PLOT X,Y
440 FOR K=1 TO 100:NEXT K
450 COLOR=0:PLOT X,Y
460 FOR K=1 TO 100:NEXT K
470 NEXT J
490 GOTO 125
```

```
600 TEXT:HOME:PRINT TAB(15);"OBSTACLE":PRINT
610 INPUT"NAME OF PLAYER ON LEFT? ";AN$
620 PRINT
630 INPUT"PLAYER ON RIGHT? ";BN$
640 J=256*PEEK(79)+PEEK(78)
650 J=RND(-J):RETURN
700 PRINT
705 FOR J=1 TO 30:C=PEEK(-16336):NEXT J
710 IF AR=0 OR BR=0 THEN 730
720 PRINT"YOU BOTH LOSE!":RETURN
730 R$=AN$:IF AR=1 THEN R$=BN$
740 PRINT R$;" WINS!"
750 RETURN
900 COLOR=A:PLOT AX,AY
910 COLOR=B:PLOT BX,BY
920 RETURN
950 COLOR=E:Y=0:FOR X=1 TO T
960 PLOT X,Y:PLOT X,Y+T:NEXT X
970 X=0:FOR Y=0 TO T
980 PLOT X,Y:PLOT X+T,Y:NEXT Y
990 RETURN
1000 IF D=1 THEN Y=Y-1
1010 IF D=2 THEN Y=Y+1
1020 IF D=3 THEN X=X-1
1030 IF D=4 THEN X=X+1
1040 R=0:IF SCRN(X,Y)<>0 THEN R=1
1050 RETURN
```

修改程式

1. 欲加速程式進行，可將 260 行的 6 改爲 4 ，若要減低速度，則可改爲 12 或 15 。
2. 欲使左右雙方於起跑時，都往上方（取代目前都以隨機的方向來跑）可插入此行：

```
168 AD=1:BD=1
```

欲使雙方都往對方的方向起跑，則可以下面一行取代：

168 AD=4:BD=3

3. 欲改變每次遊戲結束後顯示的最後訊息的時間，可修改 420 行，若想縮短顯示時間，可改 15 為 8，若欲延長，則改為 25 。

主流程

120-170	設定變數。輸入玩者名字。顯示標題及競賽場
175-205	等待使遊戲開始的鍵。重新顯示競賽場
210-250	使玩者A（左方）與玩者B（右方）移動。儲存結果
260-350	接受由鍵盤打入的鍵，將其轉換成方向
400-490	在螢幕的底部顯示贏方的名字。在碰撞發生的地方閃爍，跳回以重新玩遊戲
600-650	副程式，取得各玩者的名字
700-750	副程式，顯示贏方的名字
900-920	副程式，顯示各玩者在螢幕上所造成的障礙
950-990	副程式，顯示競賽場
1000-1050	副程式，移動玩者，並決定是否移的位置是空白

主要變數

AX,AY	玩者A目前的位置座標
BX,BY	玩者B目前的位置座標
A	玩者A的圖形顏色
B	玩者B的圖形顏色
F	讀入鍵盤的按鍵之位址(peek address)
S	鍵盤重設的位址(poke address)
T	競賽場的一邊長
AD,BD	A與B目前所走的方向(1 =上，2 =下，3 =左，4 =右)
E	競賽場周邊的顏色
C	由鍵盤按入的字
X,Y	暫時螢幕位置
D	暫時方向
AR,BR	A與B移動的結果(0 =正常，1 =碰撞)
ANS,BNS	玩者A與B的名字
Z	發生碰撞之處所顯示的顏色
J,K	迴圈變數
R$	贏方的名字

研究建議

1. 記錄連續幾局（譬如，7局）的分數，每一局結束後，顯示目前的分數。不要忘了考慮和局的情形（二者互撞）。
2. 修改程式，使遊戲雙方只能按左轉及右轉。
3. 將此遊戲改爲一人與電腦玩，而非兩個人一起玩。發展電腦的遊戲政策，使能替自己找活動空間以及切斷人類對手的活動空間。

瘋狂大賽車(ROADRACE)

經歷驚險的大賽車

目　的

想像你坐在高速的賽車內，手握方向盤馳騁在蜿蜒崎嶇的道路上。由於路況惡劣，你必須提高警覺，緊握方向盤以免撞車。在此驚險的道路上，你一天能行駛多遠？千里越野大賽車必須花幾天時間？在你的起居室就能經歷驚險萬分的大賽車。

瘋狂大賽車遊戲中最具挑戰性的就是你必須隨時依路況做緊急的應變反應。尤其它可以調整路寬以及能見度，讓你面臨不同程度的挑戰。

使用方法

程式剛開始先顯示一段彩色圖形。然後詢問你兩個輸入值：路寬，以及視綫能見度。路寬可設爲3 到12之

間。賽車的困難度就隨路寬大小而改變。窄路極難駕駛，而寬路則容易多了。視綫的能見度則分爲四等級，由“惡劣”到“良好”。當能見度良好，螢幕上的汽車位置較高。這可使駕駛人看清前面路況。若能見度差，則汽車位於螢幕較低位置，只能看到眼前的路況。

設好路寬以及視綫能見度後，瘋狂大賽車即將開始，在路的起跑綫即出現一輛汽車。然後由5開始倒數計時，當最底層的燈亮起後，呼！賽車開始！路一直繼續往螢幕上方跑，路況的彎曲急轉由程式隨機控制。你必須緊握方向盤，小心駕駛，使汽車能夠沿著路走。

汽車的方向是用鍵盤右下角的兩個鍵來控制。按左箭頭（←）可使汽車向左；按右箭頭（→）可使汽車向右。若都不按，則汽車往前走。

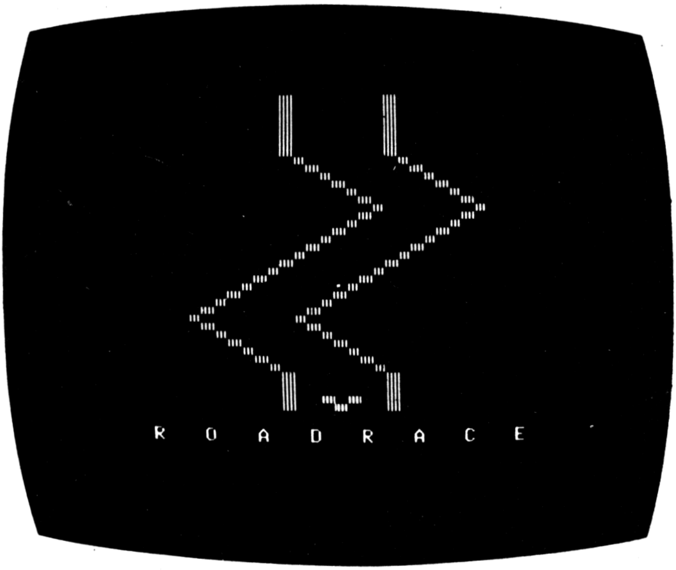
大賽車繼續進行，直到汽車“駛離路面”。每次此種撞車，就視爲一日的賽車終止。每經過一日，你可看到迄今累積的行駛哩數。

在每次撞車後，你可按C鍵，R鍵或Q鍵。選擇C鍵則可依相同的路況（即路寬及視綫）做另一天的賽程，然後記錄了累積的行車哩數。按R鍵可使賽車重新開始，此時可改變路況，然後從第一天開始算起。Q鍵則結束比賽進行回到Apple BASIC 的控制。後面兩種選擇都會顯示此次賽車每日平均的行車哩數。

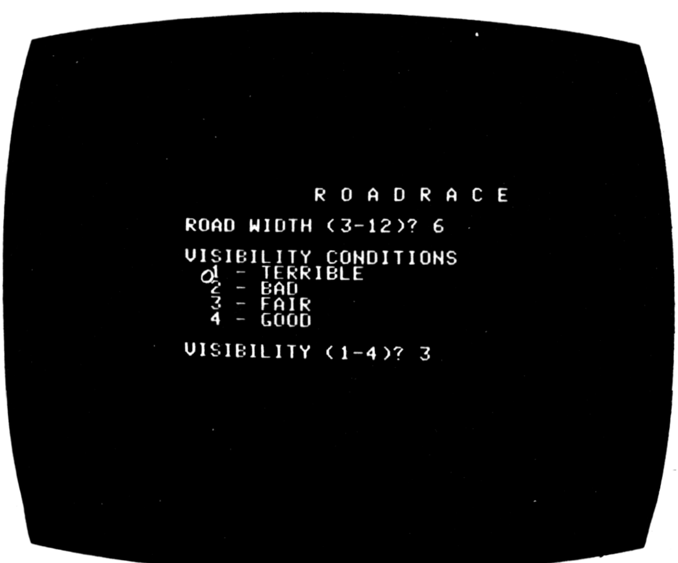
此程式有幾種不同的玩法。你可試試在特定的日數內，能行駛的最遠距離。你也可嚐試行駛34 哩最快可花多少天。和你的朋友競賽一下，看誰能夠創記錄。當你熟練了某種路況後，可將路寬訂窄或將能見度降低，

如此即能增加挑戰性。路況的差異也可判斷出個人技術的高低。

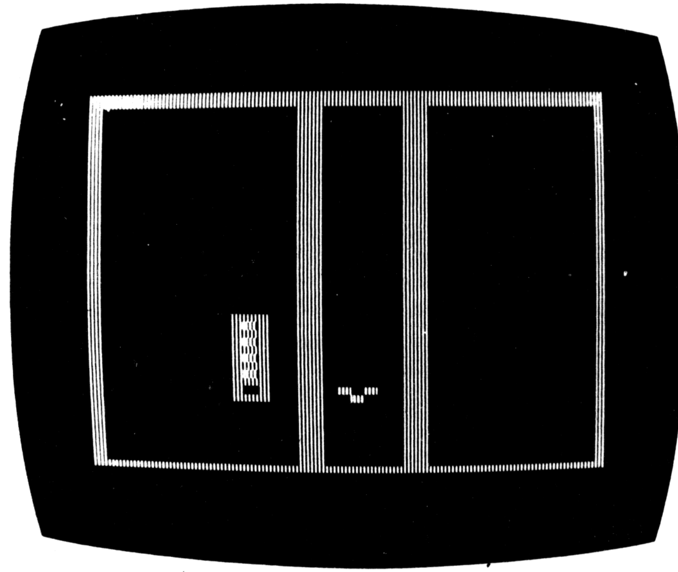
執行範例



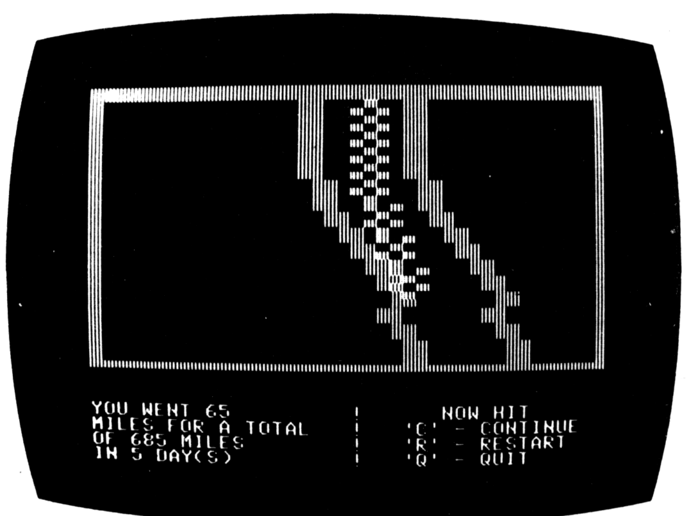
程式顯示路況簡介。見彩色部分。



操作者選擇路寬及能見度。賽車即將開始。



汽車位於起跑綫，燈倒數計時，當最下面的燈亮時，賽車開始。



操作者由鍵盤控制汽車的方向，最後撞車，在日誌記下 65 哩，5 天行程 685 哩。程　顯示選擇，由操作者決定是否繼續。

程 式

```
100 REM: ROADRACE
110 REM: COPYRIGHT 1980 BY PHIL FELDMAN AND TOM RUGG
130 U=1:Z=0:SC=-912:T8=38:F7=47
140 CR=2:CC=9
150 X=0:Y=0:KP=-16384:RK=-16368:L=0:R=0
160 LC=0.4
170 ML=136:MR=149
180 EL=1:ER=37:F5=45:F6=46:RC=1-LC
190 Q=PEEK(78)+256*PEEK(79):Q=RND(-Q):GOTO 210
200 PLOT X,Y:PLOT X-U,Y-U:PLOT X+U,Y-U:RETURN
210 GOSUB 2000:TEXT:HOME:PRINT
220 PRINT TAB(11);"R O A D R A C E"
230 PRINT:T=0:N=0
240 INPUT"ROAD WIDTH (3-12)? ";W
250 W=INT(W):IF W<3 OR W>12 THEN 240
260 PRINT:PRINT"VISIBILITY CONDITIONS"
270 PRINT"  1 - TERRIBLE"
280 PRINT"  2 - BAD"
290 PRINT"  3 - FAIR"
300 PRINT"  4 - GOOD":PRINT
310 INPUT"VISIBILITY (1-4)? ";V:V=INT(V)
320 IF V<1 OR V>4 THEN 310
330 N=N+1:L=19-W/2:R=L+W+2
340 X=20:H=0:Y=47-V*V
350 GR:POKE -16302,Z:POKE 34,U
360 POKE 33,38:POKE 32,U
370 COLOR=Z:FOR Q=40 TO F7
380 HLIN Z,39 AT Q:NEXT
390 COLOR=CR:HLIN Z,39 AT Z:HLIN Z,39 AT U
400 VLIN Z,F7 AT Z:VLIN Z,F7 AT 39
410 HLIN Z,39 AT F7
420 FOR Q=Z TO F7:HLIN L,L+U AT Q
430 HLIN R,R+U AT Q:NEXT
440 COLOR=CC:GOSUB 200
450 GOSUB 3000
500 COLOR=Z:HLIN U,T8 AT F7:CALL SC:COLOR=CR
510 HLIN U,T8 AT F7:HLIN L,L+U AT F5:HLIN R,R+U AT F5
520 H=H+U:Q=RND(U):IF Q>RC AND R<ER THEN L=L+U:R=R+U
530 IF Q<LC AND L>EL THEN L=L-U:R=R-U
540 HLIN L,L+U AT F6:HLIN R,R+U AT F6
550 COLOR=CC
600 Q=PEEK(KP):POKE RK,Z
610 IF Q=ML THEN X=X-U
```

```
620 IF Q=MR THEN X=X+U
630 IF SCRN(X,Y)=Z THEN GOSUB 200:GOTO 500
640 GOTO 1000
1000 FOR Q=U TO 6:COLOR=CC:GOSUB 200
1010 FOR J=U TO 100:NEXT:COLOR=Z:GOSUB 200
1020 FOR J=U TO 30:K=PEEK(-16336):NEXT:NEXT
1030 COLOR=CC:GOSUB 200:POKE 32,Z:POKE 33,40:CALL SC
1040 COLOR=Z:FOR Q=U TO 5:HLIN Z,39 AT F7:CALL SC
1050 NEXT:POKE -16304,0:POKE 34,20:POKE -16301,0:HOME
1060 M=H*5:T=T+M
1070 PRINT"YOU WENT ";M;:GOSUB 1300
1080 PRINT TAB(28);"NOW HIT"
1090 PRINT"MILES FOR A TOTAL";:GOSUB 1300
1100 PRINT TAB(25);"'C' - CONTINUE"
1110 PRINT"OF ";T;" MILES";:GOSUB 1300
1120 PRINT TAB(25);"'R' - RESTART"
1130 PRINT"IN ";N;" DAY(S)";:GOSUB 1300
1140 PRINT TAB(25);"'Q' - QUIT ";
1150 GET Q$:IF Q$="C" THEN 330
1160 IF Q$<>"R" AND Q$<>"Q" THEN 1150
1170 TEXT:HOME:PRINT
1180 PRINT"AVERAGE MILES PER DAY=";T/N
1190 PRINT:PRINT:PRINT:IF Q$="R" THEN 220
1200 END
1300 PRINT TAB(21);:PRINT"!";:RETURN
1310 NORMAL:RETURN
2000 GR:HOME:COLOR=CR:RESTORE
2010 X=16:Y=0:PLOT X,Y:PLOT X+8,Y
2020 FOR Y=1 TO 39:READ Q:X=X+Q-U
2030 PLOT X,Y:PLOT X+8,Y:NEXT
2040 COLOR=CC:X=20:RESTORE
2050 FOR Y=1 TO 39:READ Q:X=X+Q-U
2060 GOSUB 200:FOR J=1 TO 30:NEXT
2070 COLOR=Z:GOSUB 200:COLOR=CC:NEXT:Y=39:GOSUB 200
2080 FOR J=1 TO 25:Q=PEEK(-16336):NEXT
2090 FOR J=1 TO 6:HOME:PRINT:PRINT TAB(7);
2100 PRINT"R   O   A   D   R   A   C   E"
2110 FOR Q=1 TO 500:NEXT:HOME:PRINT:PRINT
2120 FOR Q=1 TO 250:NEXT:NEXT:RETURN
2200 DATA  1,1,1,1,1,1,1,2,2,2,2,2,2,2
2210 DATA  0,0,0,0,0,0,0,0,0,0,0,0,0,0
2220 DATA  2,2,2,2,2,2,2,1,1,1,1
3000 COLOR=CR:VLIN Y-10,Y AT L-5
3010 VLIN Y-10,Y AT L-3:FOR Q=Y-10 TO Y STEP 2
```

```
3020 PLOT L-4,Q:NEXT:FOR Q=1 TO 1000:NEXT
3030 COLOR=CC:FOR Q=Y-9 TO Y-1 STEP 2
3040 FOR J=1 TO 500:NEXT:PLOT L-4,Q:NEXT:RETURN
```

修改程式

1. 路程的彎曲程度可經由改變160行的LC值而得以調整。LC值爲0.5時，可得最大的彎曲程度。欲使道路曲度小，可將LC設小。若LC爲0，則可得到一條筆直的道路。因此，LC值必須在0與0.5之間。若想使道路的曲度大，可改160行爲：

```
160 LC=0.45
```

2. 道路的顏色以及汽車的顏色分別以CR及CL變數於140行加以設定。目前程式以深藍色爲道路的顏色，橘色爲汽車的顏色。你也許喜歡別的顏色（千萬別使用黑色，因爲其爲背景的顏色。）可參考你的Applesoft手册的低解析度彩色及其對應的數值。例如，欲改爲綠色的道路及黃色的車，可更改140行爲：

```
140 CR=12:CC=13
```

3. 操縱汽車右轉及左轉的控制鍵也可輕易地更改。若你爲左撇子，也許就想做更改；或者感覺到兩個控制鍵離遠一點，較容易操作。你修改的部分在170

行。設定適當的常數值於變數ML可與MR 即可分別設好操縱汽車左轉及右轉的控制鍵。欲找出適當的常數值，可事先執行以下的小程式。

```
10 Q=PEEK(-16384):IF Q < 128 THEN 10
20 POKE -16368,0:PRINT Q
```

執行此程式，然後按入你想做爲操縱汽車左轉的鍵，螢幕即自動顯示一數值。此數值可設入變數M L。再重新執行程式，按入你想做爲操縱汽車右轉的鍵，將螢幕顯示的數值設入變數M R 。例如，若你想以 1 鍵控制左轉， 9 鍵控制右轉。 170 行即爲：

```
170 ML=177:MR=185
```

4. 爲了加強臨場感，可用操縱桿代替鍵盤控制汽車的左轉與右轉。假設我們使用操縱桿 0 ，第一步驟就是“度量”其單位。首先執行以下的程式，

```
10 PRINT PDL(0):GOTO 10
```

手握操縱桿，調整桿的位置，直到 100 一直顯示於螢幕上。記錄此時操縱桿的位置。若操縱桿在此位置之左邊，則汽車向左轉。若操縱桿在此位置的右邊，此時螢幕顯示的值大於 100，我們可使用任何一數爲臨界值。若大於此值，則汽車向右轉。假設我們使用 160 ，做爲此右轉的臨界值，將其值設給 M R 。現在主程式就可做如下的改變。

```
170 ML=100:MR=160
600 Q=PDL(0)
610 IF Q < ML THEN X=X-U
620 IF Q > MR THEN X=X+U
```

從此就可以用操縱桿控制汽車。當操縱桿在左右兩個臨界點之間的話，汽車即能筆直往前跑。若操縱桿移到左臨界點之左邊，則汽車往左轉；若移到右操縱桿的右邊，則汽車往右轉。

主流程

130-190	設定變數
200	副程式，畫汽車
210-230	顯示遊戲簡介
240-320	操作者輸入路況（路寬及視線能見度）
330-340	設定道路
350-450	畫賽車起跑
500-550	畫下一個路面
600-640	更改汽車的位置
1000-1310	處理撞車後的事件
2000-2220	副程式，初設圖形顯像
3000-3040	副程式，顯示交通號誌

主要變數

W	路寬
V	視綫能見度
M	當日所行駛的哩數
N	大賽車迄今所經的日數
T	整個大賽車行駛的總哩數
H	賽車時所經歷的時間
ML , MR	汽車向左轉與向右轉的常數值
L , R	道路的左邊界與右邊界的位置
LC , RC	使道路往左彎及右彎的遮斷隨機值
EL , ER	道路的最左及最右允許位置
Q$	操作者的回答
X , Y	汽車的水平及垂直位置
J , K , Q	工作變數及迴圈指標
CR	道路的顏色
CC	汽車的顏色
KP	最近按入的鍵之輸入值（ peek argument ）
RK	重設鍵的接收值（ poke argument ）
SC	使螢幕捲動的呼叫引數
Z , U , T8 , F5 , F6 , F7	數值常數

研究建議

1. 寫一程序，在每次撞車後，計算駕駛員的技術。顯示一些訊息，將駕駛技術由"專家"級分到"差勁的危險駕駛員"。這種分級的計算可比較他所行駛的哩數以及對此路況的哩數平均值。若對初學者而言，你可以使用公式：

   期望哩數 $= 5W^3 + 50V - 100$

   此公式尚相當粗陋。這一段程式可放在 1110 行到 1120 行之間。

2. 也可以使兩個人同時以一輛車子比賽，然後將累積哩數分開記錄。每次撞車後，顯示目前的領先者，以及他領先的哩數。

3. 可加入一些路障或險境，以增加挑戰性。可將此部分的適當畫圖指令插入 540 行之後。若汽車撞到任何非空白的方塊，程式即馬上得知。

瓦利棋(WARI)

向電腦挑戰古非洲的瓦利棋

目　的

瓦利（Wari）棋爲起源相當早的石頭棋戲。其來源可追溯到幾千年前，其他類似的遊戲爲艾瓦利（Awari），奧威爾（Oware），帕蘭古利（Pallan-guli），卡拉（Kalah），等等。此類石頭棋通稱爲曼卡拉（Mancala）。

此程式讓你能夠與電腦對抗，玩瓦利遊戲。由於電腦是個很高竿的瓦利戲專家，因此你也許輸了好幾場之後，才能擊敗它。這似乎會稍微傷害你的自信心，因爲瓦利遊戲是純粹技術性的遊戲（就如西洋棋），不像那些雙陸棋之類的遊戲，有運氣的成份在裡面。

使用方法

程式開始執行後，首先顯示瓦利棋盤，然後詢問你是否先走。瓦利棋盤是由 12 個方格所構成，每行有 6 格共兩行。你這一方爲底邊，方格號碼由左至右爲 1 到 6 。電腦一方位於上邊，方格號碼由右至左爲 7 到 12 。

棋賽開始時，每個方格都放有 4 個石子。每顆石子都相同，可在任意的方格中移動。

第一個玩者由自己這一邊任意的一個方格中取出所有的石子，然後由其下一個號碼的方格起，依序每格放一石子。（ 12 號方格之後，就連到 1 號方格)。

若最後一顆石子（ 四顆石子之中最後放入方格的一顆 ）放入對手那邊的方格內，且使方格內的石子數爲 2 或 3 ，則移石的一方可以取出方格內的石子。而且，若最後第二顆石子所放入的方格也滿足此狀況（即在對手那一邊，且內含兩顆或三顆石子)，則移石者也可取出此方格之石，此狀況可一直倒推回來，直到對方一邊不再滿足此狀況。

不管有無取石，接著就輪到對方移石，也是依以上原則移石或取石。雙方就如此反覆輪流下去。

棋賽以最先取得 24 顆石子的一方爲勝利。 24 顆剛好是棋賽之初 48 顆石子的一半。

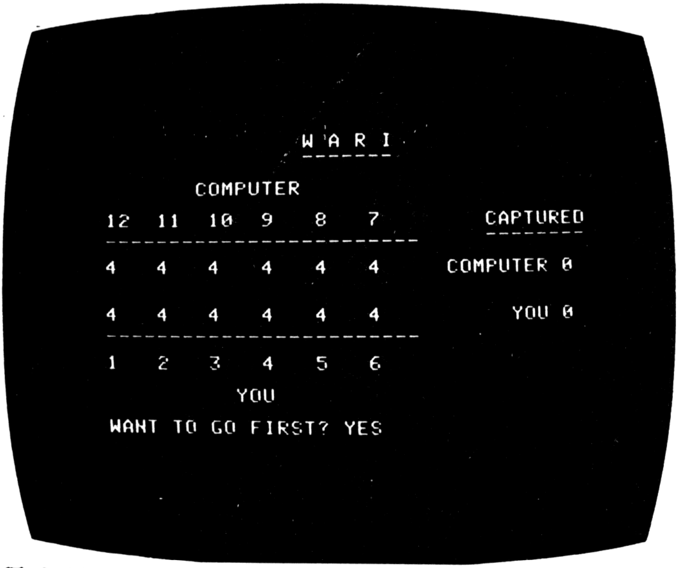
爲了應付棋賽進行之中的特殊狀況，必須訂一些特別的規則，以免比賽中斷。第一爲，不能取掉對手

那一邊方格內的所有石子。因爲如此將使對手於下一步無法移石。基於此相同的理由，因此第二爲，當你的對手那一邊無石子可資他移動（因爲他可能將他那一邊最後一顆石子移到你這一邊），你必須做適當的移石，使他那一邊至少有一顆石子可資他下一步的移動。若你無法做此移動，則棋賽只好中斷，視爲和局。

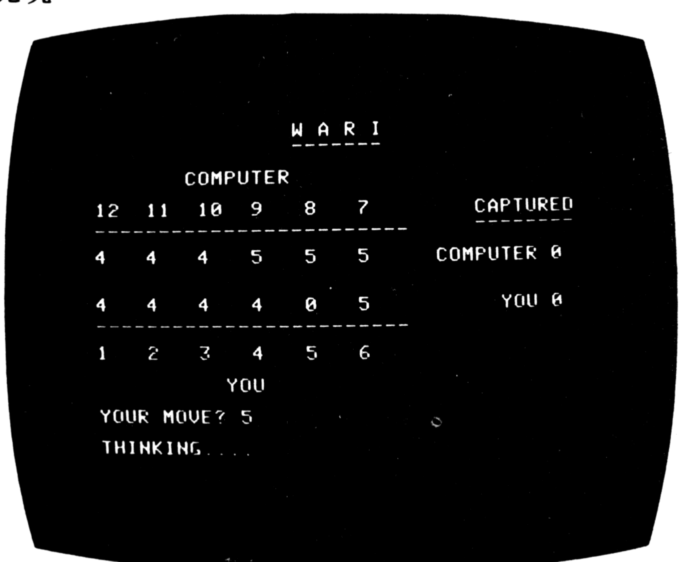
在遊戲的進行之中，一方格內可能累積了 12 顆或更多的石子。由此方格做移石，可能將這些石子分散到棋盤內所有方格。若發生此現象時，放石之時，跳過原本取石的方格，使得此方格能夠保持爲空格。

電腦做一次移石動作，依棋盤的複雜度必須花 5 秒到 30 秒之時間做思考計算，此時，螢幕即會打出“THINKING”字眼，且計算出每個可行的移法後，即會顯示一個句點（通常爲 7 到 12 的一串句點）。

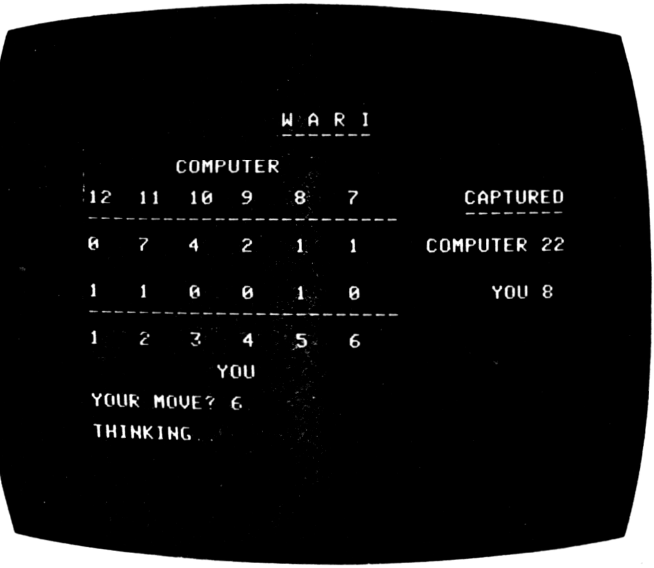
執行範例



程式開始，畫出棋盤以及詢問何以先玩。操作者決定先玩。



程式詢問操作者的移動。操作者移動方格 5，程式依此改變棋盤，然後開始思考電腦的移動。



此遊戲不久後，電腦要移方格 10 取得 4　個石子，而贏得比賽。

程　式

```
100 REM: WARI
110 REM: COPYRIGHT 1980 BY TOM RUGG AND PHIL FELDMAN
120 J=1:K=1:Q=14:P=13:F=50:D=12
130 DIM T(Q),Y(Q),W(Q),V(6),E(6),B(Q)
140 ZB=RND(1):ZB=ZB/Q:ZA=.25+ZB:ZB=.25-ZB:GOSUB 750
150 FOR J=1 TO D:B(J)=4:NEXT:B(P)=0:B(Q)=0:MN=0:GOSUB
    1200:GOSUB 900
160 GOSUB 990:INPUT"WANT TO GO FIRST? ";R$
170 GOSUB 990:PRINT D$:R$=LEFT$(R$,1):IF R$="Y" THEN
    250
180 IF R$<>"N" THEN 160
190 GOSUB 1050:PRINT D$;D$;D$;:GOSUB 1050:PRINT
    "THINKING";:GOSUB 510
195 IF M<1 THEN 2000
200 GOSUB 1050:PRINT D$;:GOSUB 1050:PRINT"MY MOVE IS ";
    M
210 FOR J=1 TO Q:T(J)=B(J):NEXT:GOSUB 350
220 FOR J=1 TO Q:B(J)=T(J):NEXT:GOSUB 900
230 IF B(Q)<24 THEN 250
240 GOSUB 1050:PRINT"I WIN!";D$:GOTO 810
```

```
250 GOSUB 990:PRINT D$;D$:GOSUB 990:INPUT"YOUR MOVE? ";
    R$
260 M=INT(VAL(R$)):IF M>6 OR M<1 THEN 330
270 FOR J=1 TO Q:T(J)=B(J):NEXT
280 GOSUB 350:IF M<0 THEN 330
290 FOR J=1 TO Q:B(J)=T(J):NEXT
300 MN=MN+1:GOSUB 900
310 IF B(P)<24 THEN 190
320 GOSUB 1050:PRINT"YOU WIN!";D$:GOTO 810
330 GOSUB 990:PRINT TAB(15);" ILLEGAL ";:FOR J=1 TO
    3000:NEXT:GOTO 250
350 IF T(M)=0 THEN M=-1:RETURN
360 R$="H":IF M>6 THEN R$="C":GOTO 380
370 FOR J=1 TO Q:Y(J)=T(J):NEXT:GOTO 400
380 FOR J=1 TO 6:Y(J)=T(J+6):Y(J+6)=T(J):NEXT
390 Y(P)=T(Q):Y(Q)=T(P):M=M-6
400 C=M:N=Y(C):FOR J=1 TO N:C=C+1
410 IF C=P THEN C=1
420 IF C=M THEN C=C+1:GOTO 410
430 Y(C)=Y(C)+1:NEXT:Y(M)=0:L=C
440 IF L<7 OR Y(L)>3 OR Y(L)<2 THEN 460
450 Y(P)=Y(P)+Y(L):Y(L)=0:L=L-1:GOTO 440
460 S=0:FOR J=7 TO D:S=S+Y(J):NEXT
470 IF S=0 THEN M=-2:RETURN
480 IF R$="H" THEN FOR J=1 TO Q:T(J)=Y(J):NEXT:RETURN
490 FOR J=1 TO 6:T(J)=Y(J+6):T(J+6)=Y(J):NEXT
500 T(Q)=Y(P):T(P)=Y(Q):RETURN
510 FOR A=1 TO 6:M=A+6:IF B(M)=0 THEN E(A)=-F:GOTO 690
530 FOR J=1 TO Q:T(J)=B(J):NEXT:GOSUB 350
540 IF M<0 THEN E(A)=-F:GOTO 690
550 IF T(Q)>23 THEN M=A+6:RETURN
560 FOR J=1 TO Q:W(J)=T(J):NEXT:FOR K=1 TO 6
570 IF T(K)=0 THEN V(K)=F:GOTO 670
580 FOR J=1 TO Q:T(J)=W(J):NEXT:M=K:GOSUB 350
590 IF M<0 THEN V(K)=F:GOTO 670
600 FA=0:FB=.05:FC=0:FD=0:FOR J=7 TO D
610 FB=FB+T(J):IF T(J)>0 THEN FA=FA+1
620 IF T(J)<3 THEN FC=FC+1
630 IF T(J)>FD THEN FD=T(J)
640 NEXT:FE=FB:FOR J=1 TO 6:FE=FE+T(J):NEXT
650 FA=FA/6:FD=1-FD/FB:FC=1-FC/6:FB=FB/FE
660 V(K)=ZA*(FA+FB)+ZB*(FC+FD)+T(Q)+B(P)-B(Q)-T(P)
670 NEXT:E(A)=F:FOR J=1 TO 6:IF V(J)<E(A) THEN E(A)=V(J)
680 NEXT
690 PRINT".";:NEXT:M=0:FA=-F:FOR J=1 TO 6
```

```
700 IF (E(J)>FA) THEN FA=E(J):M=J+6
710 NEXT:RETURN
750 A$="-":FOR J=1 TO 23:A$=A$+"-":NEXT
760 C$=CHR$(95):FOR J=1 TO 23:C$=C$+CHR$(95):NEXT
790 D$=" ":FOR J=1 TO 5:D$=D$+D$:NEXT
800 RETURN
810 PRINT"GOOD GAME!"
840 INPUT"WANT TO PLAY AGAIN? ";R$
850 R$=LEFT$(R$,1):IF R$="Y" THEN 140
860 IF R$<>"N" THEN 840
870 PRINT"SEE YOU LATER"
880 PRINT:END
900 VTAB 9
920 FOR J=0 TO 5:PRINT TAB(4*J+1);B(12-J);:IF B(12-J)=0
    THEN GOSUB 1100
930 NEXT:PRINT TAB(27);"COMPUTER ";B(Q)
940 VTAB 12:FOR J=0 TO 5
950 PRINT TAB(4*J+1);B(J+1);:IF B(J+1)=0 THEN GOSUB
    1100
960 NEXT:PRINT TAB(32);"YOU ";B(P)
970 RETURN
990 VTAB 19:HTAB 1:RETURN
1050 VTAB 21:HTAB 1:RETURN
1100 PRINT" ";:RETURN
1200 TEXT:HOME:PRINT TAB(16);"W A R I"
1210 PRINT TAB(16);"-------"
1220 PRINT:PRINT TAB(8);"COMPUTER":PRINT
1230 FOR J=0 TO 5:PRINT TAB(4*J+1);12-J;:NEXT
1240 PRINT TAB(30);"CAPTURED":PRINT C$;TAB(30);LEFT$(A$,
     8)
1250 PRINT:PRINT:PRINT:PRINT:PRINT:PRINT C$:PRINT
1260 FOR J=0 TO 5:PRINT TAB(4*J+1);J+1;:NEXT
1270 PRINT:PRINT:PRINT TAB(11);"YOU"
1280 PRINT:RETURN
2000 PRINT"NO LEGAL MOVES."
2010 PRINT"GAME IS A DRAW."
2020 GOTO 840
```

修改程式

1. 若欲使遊戲進行得快一點，且讓你的電腦對手實力弱一點，則可插入以下兩行：

```
555 GOTO 600
665 E(A)=V(K):GOTO 690
```

在此遊戲的標準版本內，電腦在計算移石法時視察了其每一步可能移動方法，以及你的可能回答。以上的改變使電腦只看他的可能移動方法，並沒檢查你的可能回答。因之，電腦的下棋功力大減，但可於幾秒內就決定下一步的移法。

2. 若你對電腦的下棋思考原則感到很好奇，可做以下的改變以探究其根源。改 690 行爲如下：

```
690 PRINT E(A);"/";:NEXT:M=0:FA=-F:FOR J=1 TO 6
```

此會使程式，顯示其每次移動所計算出來的參考數值（由 7 號方格開始）。它選定此 6 個數值的最大值做爲其下一步的動作。若爲負數值，表示電腦若做此項活動，將會失去石子（假定你可做出最好的回答）。若爲－ 50 表示其爲非法的移動。大於 1 的正值表示電腦可由此移石動作取到石子。

主流程

120-150	設定變數，顯示瓦利棋盤
160-180	詢問誰先走，處理操作者的回答
190-220	決定電腦的移石，顯示新的棋盤位置

230-240	判斷是否電腦移動導致勝利。若如此，顯示訊息
250-300	取得操作者的移石動作，檢查是否合法。顯示新的棋盤位置
310-320	判斷是否操作者的動作導致勝利
330	顯示不合法動作的訊息
350	副程式，在T陣列中做一移動M
360-390	將T陣列抄謄到Y陣列（若電腦移石，則做相反動作）
400-430	在Y陣列做一移動
440-450	檢查是否可取石。取出石子。檢查其前一個方格
460-470	判斷是否對手留下合法的移動
480-500	將Y陣列抄謄回T陣列
510-710	副程式，決定電腦的動作
750-800	副程式，建立棋盤顯像的圖形字串
810-880	顯示結束訊息，詢問是否再玩
900-970	副程式，顯示棋盤上之石子以及被取出之石
990	副程式，將螢幕標示（ cursor ）移到螢幕的“ YOUR MOVE ”位置
1050	副程式，將螢幕標示移到螢幕的“ MY MOVE ”位置
1110	副程式，顯示空白字
1200-1280	副程式，顯示瓦利棋盤（沒填入石子）
2000-2020	若電腦無法做合法的移動，則顯示訊息

主要變數

Q,P,F,D	14，13，50及12的常數值
T,Y,W	暫時抄謄瓦利棋盤的陣列
V	電腦移石後，對手的可能反應之6個動作的參考值陣列
E	電腦移石所計算出來的6個可能動作參考值陣列
B	瓦利棋盤陣列。第十三個元素爲操作者取石的數目，第十四個元素爲電腦取石的數目
ZA,ZB	估價函數的權衡因子
MN	移石的數目
R$	操作者的回答。同時也用來指示何方移石的開關（C爲電腦，H爲操作者）
M	移石動作L1－6爲操作者，7－12爲電腦），若非法則設爲負值
C	取石的方格號碼
L	置入最後一顆石子的方格號碼
S	移石後，棋盤上對方一邊的石子數
A	指示目前做電腦移石的估價計算之動作（6個可能移石動作之一）
FA	決定移石動作的第一個估價因子（電腦那一邊遭佔用的方格數）
FB	第二個估價因子（電腦那一邊的全部石

	子數）
FC	第三個估價因子（含二個或以下的石子的方格數目）
FD	第四個估價因子（電腦那一邊含石最多的方格之所含石子數）
FE	棋盤目前的石子總數
A$,C$	顯示棋盤的圖形字元字串
D$	32 個空白字的字串

研究建議

1. 修收程式，當雙方在 30 個移石動作之間無取石動作，則宣佈平手。可在 300 行使移石計數器加 1。然後追踪最近一次的取石時刻和現在的移石計數器相比較，以決定其差額是否大於 30。
2. 可試圖修改電腦決策時所用的估價函數，看是否能增強電腦下棋的功力，600 行到 660 行之間檢查考慮的移石後棋盤位置的改變情形。可實驗一下各個權衡因子，或者加入新的因子。
3. 修改程式，使能允許兩人對抗，而非與電腦對抗。

第四部分　圖形程式：

視 覺 的 享 受

簡　介

Apple 電腦爲相當吸引人的機器，因爲它具有功能完備極優良的畫圖能力。本書其他部分的有趣程式也充分利用了這些畫圖的能力對其輸出結果加添了不少趣味性。在此我們專門介紹Apple 電腦的圖形能力，及其使用於休閒娛樂的一些有趣的圖形。

小時候每當看萬花筒時，總是深深著迷於那些五彩繽紛的對稱圖形。KALEIDO萬花筒程式即能產生像萬花筒的八邊對稱圖形。

接著兩個程式可產生不同效果更具變化性的圖形。SPARKLE　星光閃爍程式可產生讓你心蕩神逸的閃爍搖動美術拼貼。SQUARES同心圖程式則使用幾何圖形組出匪夷所思的漂亮圖形。

WALLOONS空中飛人程式則表現出Apple 電腦另一種能力。此程式可讓你由電腦動畫得到極大的娛樂效果。

萬花筒(KALEIDO)

八面對稱的萬花筒

目 的

若你曾在童年看過萬花筒，應該深深地被那些無止境五彩繽紛的對稱圖形所迷惑。此程式可產生一系列類似萬花筒的圖形，一個接著一個蓋過先前的圖形。有些圖形爲對八個軸對稱，有些則對四個軸對稱——你可試著指出這些漂亮圖形的對稱軸。

使用方法

並不需多費唇舌於如何使用此程式，因只要按入RUN ，然後坐好，準備看漂亮的圖形。可將室燈關掉，放柔美的音樂。瓊音繚繞，美景在前，實爲至高無上的享受。

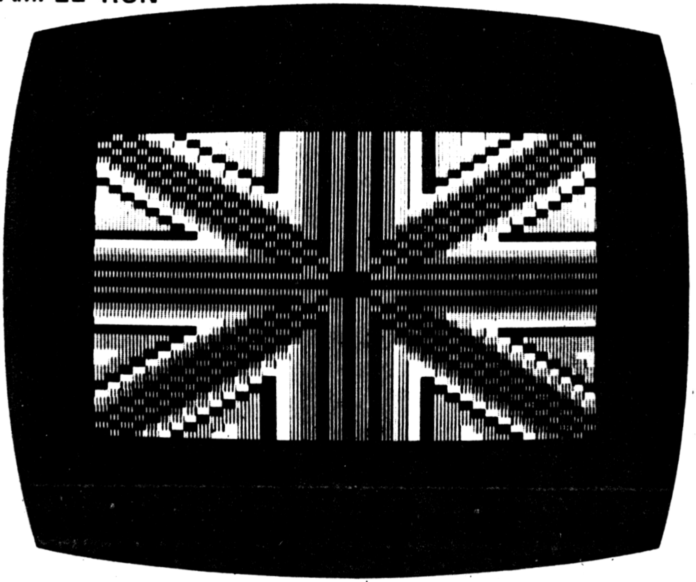
順便提及，必須修正一下剛剛所說的對稱圖形。

正確的說法應爲各個圖形字的排列位置爲對稱。老實說，整個圖形並非完全的對稱，因爲每個圖形字無法達到十足的對稱。

可叫你的朋友們一起將他們的Apple 電腦帶過來。大家同時執行萬花筒程式，看看是否能打破世界記錄。請注意，若如此一來被這些漂亮的圖形所催眠，我們一概不負責。

執行範例

SAMPLE RUN



萬花筒程式所顯示的圖形，見彩色部分

程 式

```
100 REM: KALEIDO
110 REM: COPYRIGHT 1980 BY TOM RUGG AND PHIL FELDMAN
120 HOME:GR
125 P=19
130 A=P:B=P:D=-1
135 M=15
140 DIM R(10)
150 FOR J=0 TO 10
160 R(J)=INT(M*RND(1))
170 NEXT J
180 D=-D:K=1:L=P:IF D>0 THEN 200
190 K=P:L=1
200 FOR J=K TO L STEP D
210 X=A+J:Y=B:GOSUB 900
220 X=A-J:GOSUB 900
230 X=A:Y=B+J:GOSUB 900
240 Y=B-J:GOSUB 900
250 X=A+J:Y=B+J:GOSUB 900
260 X=A-J:Y=B-J:GOSUB 900
270 Y=B+J:GOSUB 900
280 X=A+J:Y=B-J:GOSUB 900
700 NEXT J
750 FOR J=1 TO 2000:NEXT J
800 GOTO 150
900 COLOR=R(0):PLOT X,Y
910 IF J=1 THEN RETURN
920 W=INT(J/2):T=J-W-1
930 FOR N=1 TO W
940 IF X<>(A) THEN 950
945 Y2=Y:X2=X+N:GOSUB 2000:X2=X-N:GOSUB 2000:NEXT N:
    RETURN
950 IF Y<>B THEN 970
960 X2=X:Y2=Y+N:GOSUB 2000:Y2=Y-N:GOSUB 2000:NEXT N:
    RETURN
970 Y2=Y:IF X>=(A) THEN 980
975 X2=X+N:GOSUB 2000:GOTO 990
980 X2=X-N:GOSUB 2000
990 X2=X:IF Y>=B THEN 1000
995 Y2=Y+N:GOSUB 2000:GOTO 1010
1000 Y2=Y-N:GOSUB 2000
1010 NEXT N:RETURN
2000 COLOR=R(N):PLOT X2,Y2:RETURN
```

修改程式

1. 欲在下一個圖形前 20 %的時間內清除螢幕，可插入以下：

```
175 IF RND(1) <= .2 THEN GR
```

若爲 50 %，可用 .5 取代 .2 。

2. 欲隨機地改變圖形的大小，可插入：

```
177 P=INT(15*RND(1))+5
```

3. 若只想顯示外側的圖形，可改 178 行：

```
178 D=-1
```

若只想顯示內側的圖形，則改爲

```
178 D=1
```

4. 欲改變圖形使用的顏色個數，可改變 135 行的M值。必須確使其爲 2 到 16 之間的整數。

5. 欲增長每張圖形的顯示時間，可改 750 行：

```
750 FOR J=1 TO 5000:NEXT J
```

或者，刪除此行以消去其圖形間的停頓時間。

註：以上的改變可增加圖形顯示的變化性。可以一個一個試，或者多個改變一起出現。

主流程

120-140	設定變數
150-170	選擇11個隨機圖形顏色
180	反方向顯示(內側—外側)
190-700	顯示整個螢幕的圖形
800	跳回以產生下一個圖形
900-1010	畫圖形的對稱軸上面的點
2000	副程式，軸兩邊的點

主要變數

P	由圖形中心到邊的距離
A , B	圖形的中心點
D	圖形畫的方向(1 = 外側 ，- 1 = 內側)
M	決定隨機圖形顏色範圍的數值
R	11個隨機圖形顏色的陣列
J , K , L	註標變數
X , Y	軸上欲畫的點之座標
X2 , Y2	軸兩邊欲畫的點之座標
N , W	工作變數

星光閃爍(SPARKLE)

閃爍耀眼的星星

目　的

此圖形程式提供一系列的星光閃爍圖形，就如天上的繁星，向你眨著眼睛。程式內使用兩類圖形，第一類爲同中心的菱形由螢幕的中央往外擴張。雖然其圖形形狀有規則，但其產生的順序爲隨機的，因此可達到閃爍的效果。

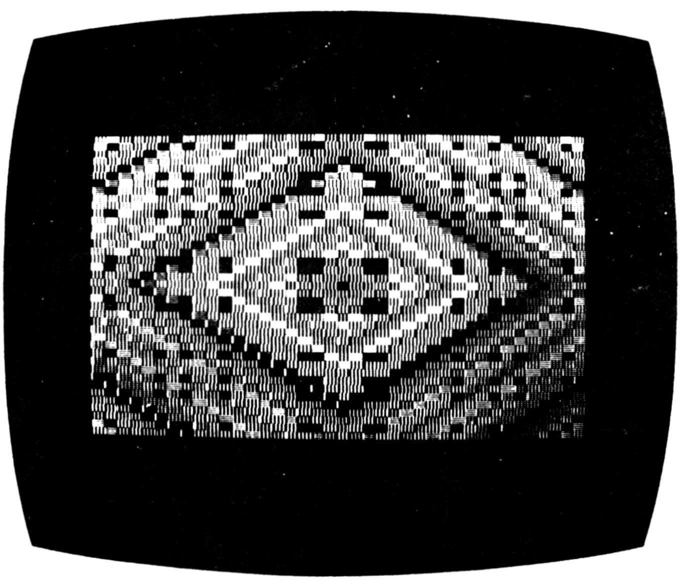
第二類圖形爲第一類圖形結束大約兩秒後才開始產生。其爲一系列的“小星星”由左至右由上到下在螢幕上跑。每個小星星的顏色爲隨機選得，而且在螢幕上的間隔相同。每一排星星的間隔也爲隨機選定。而且星星的數目也是在10顆到30顆之間隨機選得。

第二類圖形結束之後，程式跳回產生第一類圖形而蓋過第二類圖形。

使用方法

若你對剛剛的說明感到模糊不清，沒關係，你只要將程式打入Apple 電腦內，然後坐好，看看它如何表演！

執行範例



星光閃爍程式所產生的圖形，見彩色部份。

程 式

```
100 REM: SPARKLE
110 REM: COPYRIGHT 1980 BY TOM RUGG AND PHIL FELDMAN
120 HOME:GR
125 S=19
130 DIM A(S),B(S):X=19:Y=19
140 T=INT(16*RND(1))
150 FOR J=0 TO S:A(J)=J:B(J)=J:NEXT J
160 FOR J=0 TO S:R=INT((S+1)*RND(1))
170 W=A(J):A(J)=A(R):A(R)=W:NEXT J
180 FOR J=0 TO S:R=INT((S+1)*RND(1))
190 W=B(J):B(J)=B(R):B(R)=W:NEXT J
200 FOR J=0 TO S:FOR K=0 TO S
210 R=A(J):W=B(K):C=R+W+T
220 COLOR=C
240 PLOT X+R,Y+W
250 PLOT X+R,Y-W
260 PLOT X-R,Y-W
270 PLOT X-R,Y+W
280 PLOT X+W,Y+R
290 PLOT X+W,Y-R
300 PLOT X-W,Y-R
310 PLOT X-W,Y+R
320 NEXT K:NEXT J
350 FOR J=1 TO 2000:NEXT J
400 M=15
405 N=INT(21*RND(1))+10
410 FOR J=1 TO N
420 R=INT(22*RND(1))+1:W=INT(M*RND(1))
430 COLOR=W
450 FOR L=Y-S TO Y+S STEP INT(R/4)+1
460 FOR K=X-S TO X+S STEP R
470 PLOT K,L
480 NEXT K:NEXT L:NEXT J
490 GOTO 140
```

修改程式

1. 可使第二類圖形先出現。只要插入此行：

```
135 GOTO 400
```

或者，刪除第一類圖形，則插入此行：

```
145 GOTO 400
```

或者，刪除第二類圖形，則插入此行：

```
360 GOTO 140
```

2. 增加第一類圖形後的停頓時間，只要加大350行內的2000值即可，譬如，改爲5000。若刪除350行，則消除了停頓時間。
3. 增加第二類圖形的小星星數目，只要更改405行的10值改爲30或50 即可。若想減少小星星的數目，只要將10改爲1，且將405行的21改爲5或10。
4. 你可嚐試更改420行的22值改爲2到39之間的數，看看第二類圖形會發生什麼樣的改變。
5. 若想改變第二類圖形的顏色，可更改400行的M值。例如，可試：

```
400 M=4
```

必須確使M值爲2到16之間的整數。

主流程

120-130	設定變數。清除螢幕
140-320	在螢幕中央顯示菱形圖形
150-190	將 0 到 19 的數散佈到陣列A 與B 內
200-320	螢幕彩色圖形顯像
350	停頓兩秒鐘
400-480	第二類圖形，以隨機的彩色小方格一小星星蓋過螢幕

主要變數

S	第一類圖形的長度大小
R	隨機整數，工作變數
A , B	用來產生第一類圖形的陣列，其內容散佈 0 到S 之間的整數
X , Y	螢幕中央的座標（ 19 × 19 ）
T	0 到 15 間的整數，用來產生隨機圖形顏色
J , K , L	工作變數及迴圈指標
W	工作變數
C	第一類圖形所顯示的圖形顏色
N	第二類圖形的重覆次數
M	第二類圖形內，取得隨機顏色的乘數

研究建議

以另一種方式來顯示第二類圖形。本來爲由頂往下跑，可將其改爲由底往上冒。

同心圖(SQUARES)

五彩同心方形

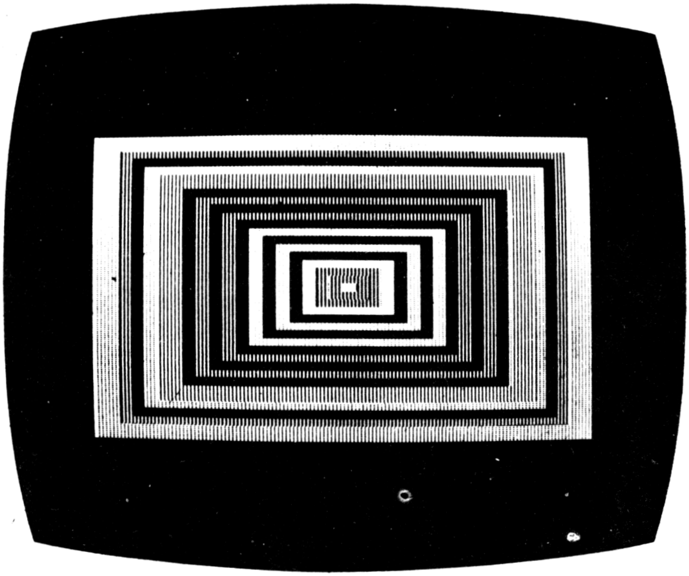
目　的

此爲另一種有趣的圖形程式。它畫出一系列的同心方形，每個方形的顏色隨機產生。一系列的同心圖填滿整個螢幕後，又重新由中心開始蓋過另一系列的同心方形。

使用方法

就如其他的圖形程式，你只要坐好，靜靜地看你的Apple 電腦如何表演即可。

執行範例



同心圖程式所產生的圖形，見彩色部分。

程 式

```
100 REM: SQUARES
110 REM: COPYRIGHT 1980 BY TOM RUGG AND PHIL FELDMAN
120 HOME:GR
130 X=20:Y=20:N=1
135 COLOR=INT(16*RND(1)):PLOT X,Y-1
140 C=INT(16*RND(1))
170 COLOR=C:FOR J=0 TO N:PLOT X+J,Y:NEXT J
180 X=X+N:N=N+1
190 FOR J=0 TO N:PLOT X,Y-J:NEXT J
200 Y=Y-N:FOR J=0 TO N:PLOT X-J,Y:NEXT J
210 X=X-N
220 FOR J=0 TO N:PLOT X,Y+J:NEXT J
230 N=N+1:Y=Y+N
240 IF N<38 THEN 140
250 FOR J=1 TO 1000:NEXT J
260 GOTO 130
```

修改程式

1. 更改每一系列同心圖顯示之後的停頓時間，只要改變 250 行的 1000 值。若值大，則停頓時間長。
2. 若欲偶而清除整個螢幕（大約所有時間的 20 %），可插入此行：

```
255 IF RND(1) < .2 THEN GR
```

主流程

120	清除螢幕
130	設定每個圖形的計數器。指標指到螢幕的中央
135	中央畫上隨機的顏色
140	選一個顏色
170	畫方形的底邊
190	畫右邊
200	畫上邊
220	畫左邊
240	檢查是否已畫到最大方形
250	停頓一秒鐘
260	跳回重新畫中心

主要變數

X，Y	畫的點的座標
N	目前所畫的邊長
C	隨機選定的顏色數值
J	廻圈變數

空中飛人(WALLOONS)

蘋果馬戲團的空中飛人

目　的

Apple 電腦是個多才多藝的機器。此程式充分運用 Apple 厲害的圖形能力以產生電腦動畫。不必驚訝，就是電腦卡通！WALLOONS空中飛人程式可讓你欣賞到蘋果馬戲團精彩的表演。

蘋果馬戲團網羅了全世界的奇人異士爲你做精彩的節目表演，讓你大開眼界。現在，就在各位的面前帶來了全世界最偉大的空中飛人！

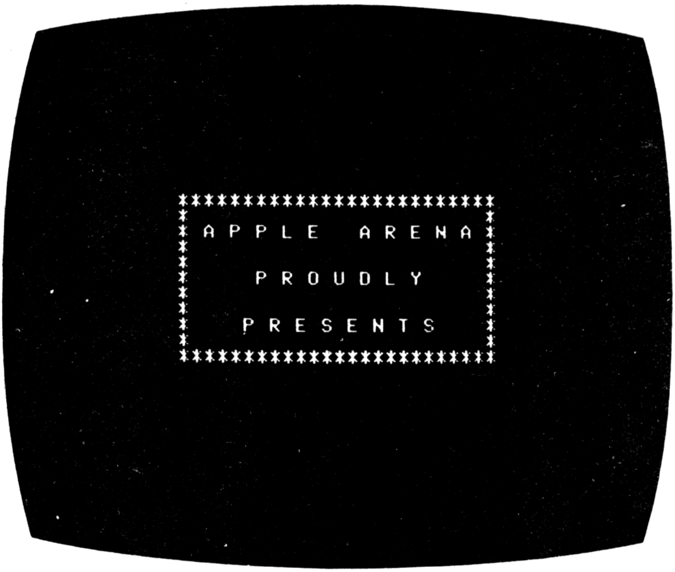
使用方法

你只要好好坐著，精神放鬆，欣賞這個精彩絕倫的表演。按入RUN 後，全世界最偉大的空中飛人即開始表演。你正好坐在最前排的中間位置，可以一覽

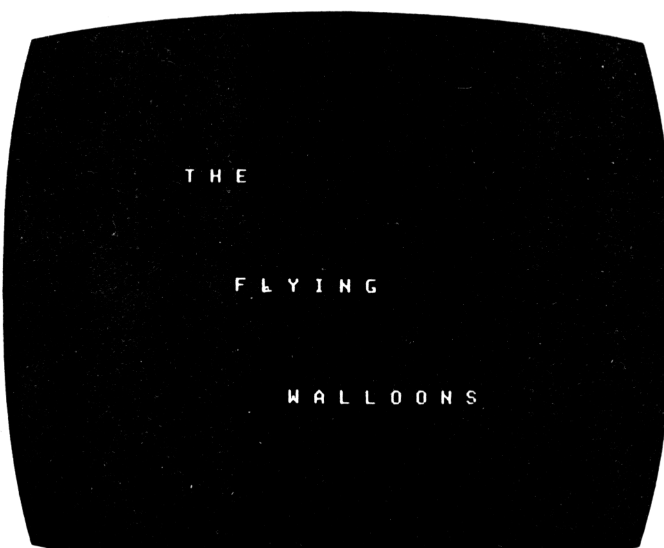
其勝況。

若你欣賞的話，請多給我們一些掌聲。

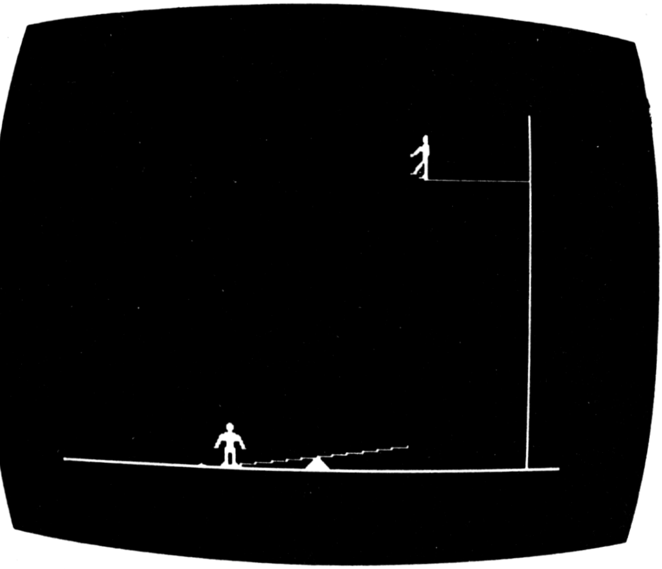
執行範例



海報貼出蘋果馬戲團的演出。



空中飛人表演開始！



空中飛人要做驚險的表演。

程 式

```
100 REM: WALLOONS
110 REM: COPYRIGHT 1980 BY PHIL FELDMAN AND TOM RUGG
120 GOTO 800
200 HCOLOR=Z:HPLOT 90,177 TO 190,189:HCOLOR=3
210 HPLOT 90,189 TO 190,177:DRAW 6 AT 140,183:RETURN
220 HCOLOR=Z:HPLOT 90,189 TO 190,177:HCOLOR=3
230 HPLOT 90,177 TO 190,189:DRAW 6 AT 140,183:RETURN
240 HPLOT 198,40 TO 258,40:HPLOT 259,5 TO 259,190:RETURN
250 XDRAW F AT X,Y:DRAW G AT X1,Y1:RETURN
260 XDRAW F AT X1,Y1:ROT=RV:DRAW G AT X,Y
270 X1=X:Y1=Y:GOSUB 500:RETURN
280 XDRAW G AT X1,Y1:DRAW F AT X,Y2:RETURN
290 XDRAW G AT X1,Y1:DRAW F AT X2,Y:RETURN
300 IF X<219 THEN FOR J=1 TO 80*(219-X):NEXT
310 RETURN
320 XDRAW F AT X,Y:DRAW G AT X,Y:RETURN
330 XDRAW G AT X,Y:DRAW F AT X,Y:RETURN
340 X=X3:Y=Y4:F=1:G=2:GOSUB 320:GOSUB 600
350 GOSUB 330:GOSUB 600:RETURN
360 DRAW F AT X3,Y3:DRAW G AT X4,Y4:RETURN
370 DRAW F AT X3,Y4:DRAW G AT X4,Y3:RETURN
```

```
400 FOR Y1=150 TO 30 STEP -5:Y=Y1+5:GOSUB 250:GOSUB 520
410 HPLOT 90,Y6 TO 190,Y7:NEXT:G=2:Y=30:Y1=35:GOSUB 250:
    F=2
420 FOR Y1=40 TO 155 STEP 5:Y=Y1-5:GOSUB 250:GOSUB 520:
    NEXT:XDRAW 2 AT X,Y3:RETURN
450 ROT=32:XDRAW F AT X4,187:DRAW G AT X4,187:ROT=Z:
    RETURN
470 FOR Y2=155 TO 55 STEP -20:Y1=Y2:Y=Y1+6:X=X6-11:RV=
    48:X1=X6
480 GOSUB 260:X=X6+1:Y=Y2+12:RV=32:GOSUB 260:X=X6+11
490 Y=Y2-4:RV=16:GOSUB 260:X=X6:Y=Y2-20:RV=Z:GOSUB 260:
    NEXT:RETURN
500 FOR J=1 TO 50:NEXT:RETURN
520 FOR J=1 TO 10:NEXT:RETURN
530 FOR J=1 TO 25:NEXT:RETURN
540 FOR J=1 TO 100:NEXT:RETURN
600 FOR J=1 TO 500:NEXT:RETURN
620 FOR J=1 TO 1000:NEXT:RETURN
700 FOR J=1 TO 25:PRINT C$;:NEXT:RETURN
720 GOSUB 530:CALL -958:RETURN
740 FOR Y=1 TO 18:HTAB X:VTAB Y:PRINT B$;:HTAB X
750 VTAB Y+1:PRINT"O";:GOSUB 500:NEXT:RETURN
800 Z=0:X=Z:Y=Z:X1=Z:Y1=Z:X2=Z:Y2=Z
810 F=Z:G=Z:X3=92:X4=188:Y3=155:Y4=167
830 B$=CHR$(32):C$="*":BL$=CHR$(7)
900 HGR2:TEXT:HOME:GOSUB 1000
910 GOSUB 5000:GOSUB 1100:HGR2
920 HCOLOR=3:ROT=Z:SCALE=1:GOSUB 1500
990 TEXT:HOME:GOSUB 4000:END
1000 X=9:Y=7:HTAB X+2:VTAB Y+2
1010 PRINT"A P P L E   A R E N A"
1020 HTAB X+6:VTAB Y+5:PRINT"P R O U D L Y"
1030 HTAB X+5:VTAB Y+8:PRINT"P R E S E N T S"
1040 HTAB X:VTAB Y:GOSUB 700:FOR J=1 TO 9:VTAB Y+J
1050 HTAB X:PRINT C$;:HTAB X+24:PRINT C$;:NEXT
1060 HTAB X:VTAB Y+10:GOSUB 700:RETURN
1100 X=9:VTAB 12:FOR K=6 TO 18 STEP 2:HTAB X+K:GOSUB
     600
1110 PRINT"-";:NEXT:FOR K=1 TO 3:GOSUB 620:NEXT:HOME:
     VTAB 5
1120 FOR X=34 TO 9 STEP -1:HTAB X:PRINT"T H E";
1130 GOSUB 720:NEXT:VTAB 12:FOR X=28 TO 13 STEP -1:HTAB
     X
1140 PRINT"F L Y I N G";:GOSUB 720:NEXT:VTAB 19
```

```
1150 FOR X=24 TO 17 STEP -1:HTAB X:PRINT
     "W A L L    N S";
1160 GOSUB 720:NEXT:GOSUB 620:GOSUB 620:X=25:GOSUB 740
1170 X=27:GOSUB 740:GOSUB 620:GOSUB 620:RETURN
1500 HPLOT Z,190 TO 279,190:HPLOT Z,191 TO 279,191
1510 GOSUB 200:GOSUB 240:X=12:Y=Y4:X1=X:Y1=Y
1520 DRAW 1 AT X,Y:DRAW 1 AT 269,Y:GOSUB 620:GOSUB 620
1530 F=1:G=3:GOSUB 250:GOSUB 620:GOSUB 620:X1=X:Y1=Y:F=
     3
1540 RV=16:X=X+32:Y=Y+11:GOSUB 260:RV=32:X=X+10:Y=Y+11
1550 GOSUB 260:RV=48:X=X+8:Y=Y-11:GOSUB 260:RV=Z:G=1:X=
     X+30
1560 Y=Y-11:GOSUB 260:PRINT BL$;BL$:GOSUB 340:X=269:Y=
     167:X1=264:Y1=Y:F=1:G=4
1570 GOSUB 620:GOSUB 620:GOSUB 250:GOSUB 600
1600 FOR Y=Y4 TO 21 STEP -6:Y1=Y-3:Y2=Y-6:X=X1:F=4:G=5
1610 GOSUB 250:GOSUB 500:GOSUB 280:GOSUB 240:GOSUB 500
1620 NEXT:Y=Y2:X=264:X1=X:Y1=Y:GOSUB 250:GOSUB 540
1630 FOR X=264 TO 201 STEP -6:X1=X-3:X2=X-6:F=5:G=4
1640 GOSUB 250:GOSUB 240:GOSUB 540:GOSUB 300
1650 GOSUB 290:GOSUB 540:GOSUB 300:NEXT
1800 GOSUB 620:GOSUB 340:FOR X=198 TO 216 STEP 6:X1=X+3
1810 X2=X+6:Y=Y2:Y1=Y:F=5:G=4:GOSUB 250:GOSUB 530:GOSUB
     290
1820 GOSUB 530:NEXT:GOSUB 620:X=X2:GOSUB 250:GOSUB 620
1830 F=4:G=4:RV=48:X=X4:Y=37:GOSUB 260:GOSUB 620
1840 ROT=Z:GOSUB 340:GOSUB 340
2000 GOSUB 600:GOSUB 620:ROT=48:RV=32:X1=X4:Y1=37
2010 X=191:Y=50:F=4:G=4:GOSUB 260:GOSUB 240:RV=16:X=190:
     Y=55:GOSUB 260
2020 RV=Z:X=189:Y=60:GOSUB 260:X1=X4:Y1=65:GOSUB 250
2030 X=X3:Y=Y4:F=1:G=2:GOSUB 320:X=X4:GOSUB 500:X1=X:F=
     4:G=4
2040 FOR Y1=70 TO Y3 STEP 5:Y=Y1-5:GOSUB 250:GOSUB 530:
     NEXT
2050 Y=Y3:X1=X3:Y1=Y4:F=4:G=2:XDRAW 4 AT X4,Y3:XDRAW 2
      AT X3,Y4
2100 FOR K=1 TO 2
2110 GOSUB 220:F=1:G=2:GOSUB 360:X=X3:X1=X:G=1:Y6=177:
     Y7=189
2120 GOSUB 400:XDRAW 2 AT X4,Y4:GOSUB 200:F=2:G=1:GOSUB
     370
2130 X=X4:X1=X:F=1:Y6=189:Y7=177:GOSUB 400:XDRAW 2 AT
     X3,Y4:NEXT
```

```
2300 GOSUB 220:F=3:G=2:GOSUB 360:G=3
2310 X6=X3:GOSUB 470:FOR Y2=35 TO 135 STEP 20:Y1=Y2:Y=
     Y2+16
2320 X1=X3:X=X1-11:RV=48:GOSUB 260:RV=32:X=X3+1:Y=Y2+32:
     GOSUB 260
2330 RV=16:X=X3+11:Y=Y2+26:GOSUB 260:RV=0:X=X3:Y=Y2+20:
     GOSUB 260:NEXT
2340 XDRAW F AT X3,Y3:XDRAW 2 AT X4,Y4:GOSUB 200:F=2:G=
     3:GOSUB 370:F=3
2350 X6=X4:GOSUB 470:X1=X4:Y1=35:RV=48:X=177:Y=51:G=2
2360 GOSUB 260:RV=32:X=189:Y=66:F=2:G=3:GOSUB 260:F=3
2370 G=1:FOR Y=66 TO 166 STEP 10:Y1=Y+5:Y2=Y+10:GOSUB
     250
2380 GOSUB 500:GOSUB 280:GOSUB 500:NEXT
2390 XDRAW F AT X,Y2:ROT=0:XDRAW 2 AT X3,Y4
2400 GOSUB 220:ROT=32:DRAW 2 AT X4,187:ROT=0:DRAW 1 AT
     X3,154:PRINT BL$
2800 X=X3:F=1:G=1:FOR Y=154 TO 14 STEP -10:Y1=Y-5:X1=X+
     3:GOSUB 250:GOSUB 530
2810 X=X1+3:Y2=Y1-5:GOSUB 280:GOSUB 530:NEXT:G=2:X=182
2820 Y=4:X1=X4:Y1=8:GOSUB 250:GOSUB 500:F=2:X=194
2830 Y2=12:GOSUB 280:GOSUB 500:Y=Y2:X1=200:Y1=17
2840 GOSUB 250:GOSUB 620:F=1:GOSUB 280:GOSUB 500:G=1
2850 X1=X4:Y1=8:GOSUB 250:GOSUB 500:X=182:Y2=4:F=2
2860 GOSUB 280:GOSUB 500:G=2:FOR Y=4 TO 144 STEP 10:F=2
2870 Y1=Y+5:X1=X-3:GOSUB 250:G=3:GOSUB 450:X=X1-3
2880 Y2=Y1+5:G=2:GOSUB 280:F=3:GOSUB 450:NEXT
3000 XDRAW 2 AT X3,154:ROT=32:XDRAW 2 AT X4,187:ROT=0:
     GOSUB 200
3010 DRAW 2 AT X3,Y4:ROT=32:DRAW 2 AT X4,176:X=X4
3020 FOR Y=176 TO 56 STEP -10:X1=X-5:Y1=Y-5:F=2
3030 G=3:GOSUB 250:GOSUB 500:X=X1-5:Y2=Y-10:GOSUB 280
3040 GOSUB 500:NEXT:X=58:X1=X:Y=46:Y1=51:G=1
3050 GOSUB 250:F=1:FOR Y=51 TO 181 STEP 5:Y1=Y+5
3060 GOSUB 250:GOSUB 520:NEXT:Y2=189:GOSUB 280
3070 FOR Y=189 TO 169 STEP -5:Y1=Y-5:GOSUB 250:GOSUB
     530
3080 NEXT:FOR Y=164 TO 184 STEP 5:Y1=Y+5:GOSUB 250
3090 GOSUB 530:NEXT:PRINT BL$
3500 GOSUB 600:Y=187:XDRAW 1 AT X,189:ROT=16
3510 DRAW 4 AT X,Y:GOSUB 620:F=4:G=5:GOSUB 320
3520 GOSUB 620:GOSUB 330:GOSUB 620:ROT=0:X=X3:Y=Y4
3530 F=2:G=4:GOSUB 320:GOSUB 620:GOSUB 600:F=1:GOSUB
     330
3540 GOSUB 620:GOSUB 600:G=2:GOSUB 320
```

```
3550 PRINT BL$;BL$;BL$:GOSUB 620:RETURN
4000 HTAB 17:VTAB 10:PRINT"F I N I S"
4010 GOSUB 620:GOSUB 620:HTAB 1:VTAB 20:RETURN
5000 X6=25600
5010 RESTORE:F=INT(X6/256):G=X6-F*256:POKE 232,G:POKE
     233,F:X=X6-1
5100 READ Y:IF Y=999 THEN 5300
5110 IF Y>=0 THEN X=X+1:POKE X,Y:GOTO 5100
5120 READ F:FOR J=1 TO -Y:X=X+1:POKE X,F:NEXT
5130 GOTO 5100
5300 FOR J=208 TO 278:Y=PEEK(X6+J):POKE X6-100+J,Y:NEXT
5400 FOR J=79 TO 107:Y=PEEK(X6+J):POKE X6+100+J,Y:NEXT:
     RETURN
6000 DATA  6,0,14,0,108,0,208,0,69,1,109,1,175,1
6010 DATA  173,63,55,45,53,63,119,109,58,63,63
6020 DATA  159,-4,45,173,-5,63,55,-5,45,173,-6,63
6030 DATA  55,109,9,45,109,9,173,63,223,59,63,223,59,55,
     109
6040 DATA  73,45,109,73,173,63,223,59,63,63,223,59,119
6050 DATA  73,9,45,45,53,63,63,55,109,41,62,223,55
6060 DATA  109,41,62,223,55,109,41,21,63,223,63,23,45
6070 DATA  77,9,45,5,-101,0
6100 DATA  173,63,255,219,219,42,77,73,41,45
6110 DATA  77,73,41,30,63,223,27,63,255,219
6120 DATA  63,46,109,73,41,77,73,45,30,63,223,63,63
6130 DATA  255,59,55,-7,45,245,-6,63,119,-5,45
6140 DATA  245,59,-3,63,78,45,53,63,55,45,173,63,63
6150 DATA  55,45,45,173,-3,63,55,45,77,45,21,63,223,27
6160 DATA  63,23,45,77,73,41,173,63,223,219,219,63,46
6170 DATA  109,73,73,41,53,255,-3,219,255,42,109
6180 DATA  -3,73,9,45,5,0
6200 DATA  53,63,46,53,63,14,173,63,191,45,45
6210 DATA  62,63,55,45,45,62,63,55,45,45,30,63,46
6220 DATA  53,63,46,53,55,53,55,53,55,53,63,159,45,45,5,
     0
6300 DATA  53,63,46,53,63,14,173,63,255,42
6310 DATA  45,45,62,63,63,191,-3,45,53
6320 DATA  63,255,59,255,42,45,77,41,45,30
6330 DATA  63,223,219,119,73,73,45,62,191,45
6340 DATA  53,255,191,45,13,53,255,59,55,109
6350 DATA  9,53,255,219,255,42,45,77,9,53,63
6360 DATA  159,45,45,5,0
6400 DATA  21,63,23,45,45,21,-3,63,23
6410 DATA  -4,45,21,-5,63,23,-6,45,5,0
6420 DATA  999
```

修改程式

1. 若你想讓空中飛人在表演時多做（或少做）一些跳躍，可改變2100行的迴圈限制2。若想做4個跳躍，可使用：

```
2100 FOR K=1 TO 4
```

2. 目前標題海報是以星號爲花邊。若欲使用別的符號，可改830行的C$ 字串值。例如，若以加號爲花邊，則改830行爲：

```
830 B$=CHR$(32):C$="+":BL$=CHR$(7)
```

3. 你也許想參與空中飛人的演出，而使自己的名字出現於海報上。可直接修改1010行的字串"APPLE ARENA　"。然而你的字串必須少於22個字母，否則後面的字母將會被切除。例如，Simon Q. Fenster 先生要表演空中飛人，則改1010行爲：

```
1010 PRINT"MR. SIMON Q. FENSTER"
```

4. 目前的程式使用了高解析度的圖形，且將形狀列表儲存於位址（十進位）25600內。若你的系統有16K的RAM且Applesoft 在ROM內（見附錄1），你就必須對程式做如下的修正。

```
 900 HGR:TEXT:HOME:GOSUB 1000
 910 GOSUB 5000:GOSUB 1100:HGR:POKE -16302,0
5000 X6=7690
```

主流程

200-490	副程式，畫（及擦掉）空中飛人以及其表演的姿式
500-620	時間停頓副程式
700-750	公用副程式
800-830	設定變數
900-990	控制文字及圖形顯示
1000-1060	副程式，顯示海報
1100-1170	宣佈節目開始
1500-1570	空中飛人進場
1600-2880	空中飛人開始表演
300-3550	空中飛人表演結束
4000-4010	顯示謝幕詞
5000-6420	副程式，儲存形狀列表於位址X 6 處

主要變數

X，Y	目前空中飛人的位置。也做爲形狀列表的資料值及位置

X1,Y1; X2,Y2; X3,Y3; X4,Y4	空中飛人的參考位置
F,G	目前畫(或擦掉)的形狀;也做爲形狀列表的註標值
X6	形狀列表的起始位址
RV	圖形旋轉因子
K,J	廻圈指標
B$	空白字元
C$	海報的字串
BL$	搖鈴的ASCII 值
Z	零

研究建議

1. 有好幾種方式可加添空中飛人表演的花式。也許你想改變其結束的情形。爲了讓你能動動腦筋,以下爲另一種結束表演的方式:

```
3020 FOR Y=176 TO 96 STEP -10:X1=X-5:Y1=Y-5:F=2
3040 GOSUB 500:NEXT:Y=Y2:X1=X-5:Y1=Y-5:GOSUB 250
3050 F=3:X=X1:FOR Y=81 TO 161 STEP 5:Y1=Y+5
3060 GOSUB 250:GOSUB 520:NEXT:GOTO 3550
```

2. 若你加入其他技巧,或者依上一題的方式結束表演,試圖讓空中飛人的結束表演做隨機式的改變

，使每次程式執行的時候空中飛人的表演各不相同（至少其結束方式不同）。

3. 自行收集擴充蘋果馬戲團的各項表演。也許有一天你能建立一個完整的軟體表演圖書館。

第五部分 數學程式：

數學家、工程師與統計師的助手

簡 介

自從電腦的發明以來，電腦主要就是用來解決數學的問題。由於它們的計算速度驚人，可信度極高，可解甚多極困難，甚至人類無法完成的計算。有很多數值方法可使電腦很容易地解決一些難題。下列的程式爲其典型的例子。這些程式可提供給工程師，學生，數學家，統計師，以及在其工作之中碰到此類問題的人們使用。

GRAPH 圖表程式充分利用了Apple 電腦的畫圖能力，以畫出函數圖Y = f (x)。此函數可由你提供輸入。INTEGRATE 積分程式計算積分值，或者任何函數曲線內的面積。

實驗數據的工作常產生一堆X 與Y 值的數據。CURVE 曲線程式幫你找出一個多項式以趨近這些數據。

理論科學家（或數學系的學生）往往必須解綫性聯立方程式，可交給SIMEQN聯立方程式程式來做。

現代工程師的工作常必須解微分方程式。DIFF-EQN 微分方程式程式可幫你解一般一次微分方程式。

STATS 統計程式可輸入一列表的資料，然後計算其標準方差。而且，它可依數值的大小分類此一資料列表。

曲線(CURVE)

曲線趨近

目　的

CURVE 曲綫程式可用一個多項式函數來趨近一組資料。資料的格式必須爲X－Y 對的點。一般的實驗數據或者參考工具書內的樣本圖表資料常常都是此類型態的資料。

有很多原因促使你需要一個分析函數來表示你的資料內所顯現的函數關係。通常你的實驗數據上的Y值具有一些實驗誤差。一個優良的方程式通常可緩和這些誤差。也許你想知道在實驗中沒獲得的X值其所對應的Y值。這可能爲已知兩個X值之間的數據（可用內插法獲得），或者超出實驗範圍的數據（由外插法獲得）。若你想使用電腦程式內的資料，一個優良的方程式是相當方便及有效的方法。

此程式可用下列形式的曲綫：

$$Y = C_0 + C_1X^1 + C_2X^2 + \ldots + C_DX^D$$

趨近你的資料。你可選擇最高次方D直到7。常數係數C_0到C_D爲程式的主要輸出。而且還計算了其趨近的狀況，以便你對其正確性做個參考。你也許想要以相同的資料取得不同次方的多項式，或者對特定的X值詢問其對應的Y值。

在此類計算所使用的數值方法稱爲最小平方曲綫趨近（least squares curve fitting）。它取誤差的平方和的最小值。此最低平方法將其問題代換爲一組聯立方程式的問題。因此這些方程式可用SIMEQN 聯立方程式程式內的演算法來加以解答。事實上，一旦建立好適當的方程式，CURVE曲綫程式使用與SIMEQN 聯立方程式程式內相同的副程式來解這些方程式。若欲更深入的研究，可看參考書目所列的數值方法專書。

使用方法

你該做的第一件事當然就是將資料輸入程式內。其工作爲打入一對一對的數值。每一對數值含一個X值及其相對應的Y值。此對數值以逗點分開。一個問號即爲催促你輸入一對資料。當你將所有資料輸入後，按入

999,999

以指示資料的結束。當你打完這些後，程式會回答指出輸入的資料對數目。目前最多允許 75 對資料。

接著，你必須輸入趨近函數的最大次方，此爲 0 到 7 的整數。而且次方D 必須低於資料對的數目。

以下爲選擇D 該注意的原則：若$D = 0$，程式即輸出Y 值的平均值做爲係數C_0，若$D = 1$，程式即計算通過這些資料的最佳直綫。此種特殊現象稱爲綫性回歸（ linear regression ）。若D 值爲資料對數目減 1，程式即能找出資料正確的趨近（不計那些四捨五入的誤差及其他數值誤差）。此一解答爲通過這些資料點的曲綫。

一旦你輸入所需的次方後，程式開始計算其結果。在程式計算之時，會有一段停頓時間。此段時間的長短依資料對的個數及選定的次方大小而定。若爲25 對資料且 3 次方趨近，停頓時間大約在半分鐘以內；50 對資料且 5 次方趨近，則停頓時間約爲一分鐘。

計算結果顯示在一個列表內，其爲由 0 到D 的X 次方之係數，亦即C_0 到C_D 的值。而且還顯示了趨近的百分比。由此用來度量程式能夠趨近的正確情形。若爲 100 %表示其爲正確趨近我們在此很難斷言何值可代表很滿意的趨近，因爲其依資料的可靠度以及使用的目的而定。但也有個基本原則，若趨近度爲 90 幾，已經爲相當優良的趨近了。爲一些有求知慾的讀者提供一個計算趨近度的方程式：

$$趨近度 = 100 * \sqrt{1 - \frac{\sum_i (Y_i - Y_i)^2}{\sum_i (Y_i - \bar{Y})^2}}$$

其中Y_1為實際的Y資料值。Y_1為計算的Y值(經由多項式算出來的),而$\bar{Y}$為Y的平均值。

程式接著讓你立即面臨三個選擇,其為(1)計算特定的點。(2)以別的次方做趨近。(3)結束程式。你的選擇只要打入1,2,或3即可。各個選擇的描述如下:

選擇1允許你依目前的趨近多項式來看特定的X值所對應的Y值。在此狀況下,程式會一直催促你輸入任意X值。然後它會顯示其所對應的Y值。若欲結束此狀態,只要輸入X值為999即可。

選擇2允許你對於相同的資料以別的次方的多項式來加以趨近。通常你都想以較高的D值來趨近以增加其趨近度,除非產生嚴重的捨去誤差,較高的次方通常增加其趨近度。

選擇3只是終止程式的執行。

範例問題

問題:一個藝術收藏家想買布拉姆(Primo)的傑作"結霜的幻想"(Frosted Fantasy)。從1940年起,此幅油畫已被拍賣了七次。以下即為每次拍賣的價格記錄。

Year	*Price*
1940	$ 8000.
1948	$13000.
1951	$16000.
1956	$20000.
1962	$28000.
1968	$39000.
1975	$53000.

此油畫要在1981 年再度拍賣。此收藏家應花多少錢才可以買回此幅油畫？若他在1985 年將此幅畫賣出，預計他可得多少利益。

解答：此收藏家應該找出一個趨近多項式，以年代爲X值，價錢爲Y值。此正好適合CURVE 曲綫程式。欲使各數的單位變小一點，年代就從1900 年開始算起，而價錢則以 $1000 爲單位（因此，40 代表1940年，價錢 8 代表 $8000 ）。

執行範例

```
   - LEAST SQUARES CURVE FITTING -

 ENTER A DATA PAIR IN RESPONSE TO EACH
 QUESTION MARK.  EACH PAIR IS AN X VALUE
 AND A Y VALUE SEPARATED BY A COMMA.

 AFTER ALL DATA IS ENTERED, TYPE
  999,999
 IN RESPONSE TO THE LAST QUESTION MARK.

 THE PROGRAM IS CURRENTLY SET TO
 ACCEPT A MAXIMUM OF 75 DATA PAIRS.

 X,Y=? 40,8
 X,Y=? 48,13
 X,Y=? 51,16
```

```
X,Y=? 56,20
X,Y=? 62,28
X,Y=? 68,39
X,Y=? 75,53
X,Y=? 999,999

7 DATA PAIRS ENTERED

DEGREE OF POLYNOMIAL TO BE FITTED? 1

X POWER          COEFFICIENT
-------          -----------
   0             -48.2701205
   1             1.28722711

PERCENT GOODNESS OF FIT= 97.5302068

-- CONTINUATION OPTIONS --

  1 - DETERMINE SPECIFIC POINTS
  2 - FIT ANOTHER DEGREE TO SAME DATA
  3 - END PROGRAM

WHAT NEXT? 2

DEGREE OF POLYNOMIAL TO BE FITTED?2

X POWER          COEFFICIENT
-------          -----------
   0             38.475481
   1             -1.83492574
   2             .0270347151

PERCENT GOODNESS OF FIT= 99.9485752
                        :
   (continuation options displayed again)
                        :
WHAT NEXT? 1

ENTER 999 TO LEAVE THIS MODE

X=? 81
Y= 67.2212621
```

```
X=? 85
Y= 77.8326099

X=? 999
                          ⋮
   (continuation options displayed again)
                          ⋮
WHAT NEXT? 3
```

剛開始，使用1次方的趨近，得到97.5％ 的趨近度。收藏家想獲得更好的答案，因此就嚐試2次方趨近，此時即獲得極佳的趨近度。然後他詢問 1981 與1985的外插資料值。發現在1981年，他必須花費 $67200 以購得此油畫。若在1985 年賣出的話，預計可得 $10500 的利益。

當然，收藏家不應單獨依據此程式來幫他做決定。此結果只能當作他做決定的參考資料。並不做任何保證在未來的財務發展狀況一定與過去類似。雖然CURVE 曲綫程式爲極佳的方法，但外插資料的可靠度往往會遭到懷疑的。

程 式

```
100 REM: CURVE
110 REM: COPYRIGHT 1980 BY PHIL FELDMAN AND TOM RUGG
130 TEXT
150 MX=75
160 EF=999
170 MD=7
```

```
200 DIM X(MX),Y(MX)
210 Q=MD+1:DIM A(Q,Q),R(Q),V(Q)
220 Q=MD*2:DIM P(Q)
300 HOME:PRINT"  - LEAST SQUARES CURVE FITTING -":PRINT
310 PRINT"ENTER A DATA PAIR IN RESPONSE TO EACH"
320 PRINT"QUESTION MARK.  EACH PAIR IS AN X VALUE"
330 PRINT"AND A Y VALUE SEPARATED BY A COMMA.":PRINT
340 PRINT:PRINT"AFTER ALL DATA IS ENTERED, TYPE"
350 PRINT SPC(1);EF;",";EF
360 PRINT"IN RESPONSE TO THE LAST QUESTION MARK.":PRINT
370 PRINT:PRINT"THE PROGRAM IS CURRENTLY SET TO"
380 PRINT"ACCEPT A MAXIMUM OF ";MX;" DATA PAIRS."
400 PRINT:PRINT:J=0
410 J=J+1:INPUT"X,Y=? ";X(J),Y(J)
420 IF X(J)=EF AND Y(J)=EF THEN J=J-1:GOTO 450
430 IF J=MX THEN PRINT:PRINT"NO MORE DATA ALLOWED":GOTO
    450
440 GOTO 410
450 NP=J:PRINT
460 IF NP=0 THEN GOSUB 1600:PRINT"NO DATA ENTERED":STOP
470 PRINT NP;" DATA PAIRS ENTERED":PRINT
500 PRINT:INPUT"DEGREE OF POLYNOMIAL TO BE FITTED? ";D:
    PRINT
510 IF D<0 THEN GOSUB 1500:PRINT"DEGREE MUST BE >= 0":
    GOTO 500
520 D=INT(D):IF D<NP THEN 540
530 GOSUB 1500:PRINT"NOT ENOUGH DATA":GOTO 500
540 D2=2*D:IF D>MD THEN GOSUB 1500:PRINT
    "DEGREE TOO HIGH":GOTO 500
550 N=D+1
600 FOR J=1 TO D2:P(J)=0:FOR K=1 TO NP
610 P(J)=P(J)+X(K)^J:NEXT:NEXT:P(0)=NP
620 R(1)=0:FOR J=1 TO NP:R(1)=R(1)+Y(J)
630 NEXT:IF N=1 THEN 660
640 FOR J=2 TO N:R(J)=0:FOR K=1 TO NP
650 R(J)=R(J)+Y(K)*X(K)^(J-1):NEXT:NEXT
660 FOR J=1 TO N:FOR K=1 TO N:A(J,K)=P(J+K-2):NEXT:NEXT
670 GOSUB 2000
700 PRINT:PRINT TAB(1);"X POWER         COEFFICIENT"
710 PRINT TAB(1);:FOR J=1 TO 7:PRINT"-";:NEXT:PRINT
    TAB(17);
720 FOR J=1 TO 11:PRINT"-";:NEXT:PRINT
730 FOR J=1 TO N:PRINT"   ";J-1,V(J):NEXT:PRINT:PRINT
740 Q=0:FOR J=1 TO NP:Q=Q+Y(J):NEXT:M=Q/NP:T=0:G=0:FOR
    J=1 TO NP
```

```
750 Q=0:FOR K=1 TO N:Q=Q+V(K)*X(J)↑(K-1):NEXT:T=T+(Y(J)-
    Q)↑2
760 G=G+(Y(J)-M)↑2:NEXT:IF G=0 THEN T=100:GOTO 780
770 T=100*SQR(1-T/G)
780 PRINT"PERCENT GOODNESS OF FIT= ";T
800 PRINT:PRINT"-- CONTINUATION OPTIONS --":PRINT
810 PRINT"  1 - DETERMINE SPECIFIC POINTS"
820 PRINT"  2 - FIT ANOTHER DEGREE TO SAME DATA"
830 PRINT"  3 - END PROGRAM":PRINT
840 INPUT"WHAT NEXT? ";Q:Q=INT(Q):IF Q=3 THEN END
850 IF Q=2 THEN 500
860 IF Q<>1 THEN 800
900 PRINT:PRINT:PRINT"ENTER ";EF;" TO LEAVE THIS MODE"
910 PRINT:INPUT"X=? ";XV:IF XV=EF THEN 800
920 YV=0:FOR K=1 TO N
930 YV=YV+V(K)*XV↑(K-1):NEXT:PRINT"Y= ";YV
940 GOTO 910
1500 PRINT"** ";:FLASH:PRINT"ERROR!";:NORMAL
1510 PRINT" ** -- ";:RETURN
1600 PRINT"** ";:FLASH:PRINT"FATAL ERROR!";:NORMAL
1610 PRINT" ** -- ";:RETURN
2000 IF N=1 THEN V(1)=R(1)/A(1,1):RETURN
2010 FOR K=1 TO N-1
2020 I=K+1
2030 L=K
2040 IF ABS(A(I,K))>ABS(A(L,K)) THEN L=I
2050 IF I<N THEN I=I+1:GOTO 2040
2060 IF L=K THEN 2100
2070 FOR J=K TO N:Q=A(K,J):A(K,J)=A(L,J)
2080 A(L,J)=Q:NEXT
2090 Q=R(K):R(K)=R(L):R(L)=Q
2100 I=K+1
2110 Q=A(I,K)/A(K,K):A(I,K)=0
2120 FOR J=K+1 TO N:A(I,J)=A(I,J)-Q*A(K,J):NEXT
2130 R(I)=R(I)-Q*R(K):IF I<N THEN I=I+1:GOTO 2110
2140 NEXT
2150 V(N)=R(N)/A(N,N):FOR I=N-1 TO 1 STEP -1
2160 Q=0:FOR J=I+1 TO N:Q=Q+A(I,J)*V(J)
2170 V(I)=(R(I)-Q)/A(I,I):NEXT:NEXT
2180 RETURN
```

修改程式

1. 程式使用999做為標示各種輸入狀況的結束。若你的資料含有999的值時，即面臨一個難題。你可輕易地修改此標示值，只要改變160行的EF值不與你輸入的資料相衝突即可。例如，欲使用10101，則可做以下的修改：

```
160 EF=10101
```

2. 目前程式允許的資料對的最大個數為75 。若你需較多的資料，可改150行的MX值。例如，允許200對資料，則可使用：

```
150 MX=200
```

3. 若要允許大於7次方的趨近，可設170行內的ML值為所需的最大次方。例如，若允許10 次方的趨近，則改為：

```
170 MD=10
```

然而，在此必須強調，高次方的趨近，也許會帶來不可靠的趨近，除非你的資料特性極佳（X及Y值靠近1），若D值大於5。以後，程式通常無法產生正確的結果。此乃因為X次方的和與Y值的計算都到達2*D 次方。這些不同等級的數相加或其他運算，會產生大量的捨去及四捨五入

誤差。因此ML實用的極限爲7。

4. 你若在"程式擴充"前兩題中增加了MX值或MD值，則必須增加RAM 記憶體的可用空間。目前的程式可在Apple 含4K byte 可用空間的機器上執行（見附錄1）。若你的系統的可用空間不夠，就必須減低MX或MD的值。若你的MX或MD過大，則執行後就會發生超出記憶(out-of-memory)或不正確數值(illegal-quantity)的錯誤。

主流程

150-170	設定變數
200-220	聲明陣列維數
300-380	顯示簡介訊息
400-470	操作者輸入X－Y資料對
500-550	操作者輸入多項式的次方，判斷其是否合法
600-670	設立聯合方程式，呼叫副程式以解方程式
700-780	計算趨近度的百分比，顯示所有結果
800-860	輸入操作者的選擇，跳到適當的位置執行
900-940	對於任意X值，算出對應的Y值
1500-1610	副程式，印出錯誤訊息

2000-2180 副程式，解綫性聯立方程式

主要變數

MX	可容納的最大資料對個數
MD	容許的趨近最大次方
EF	資料輸入的結束標示值
X,Y	X與Y資料對的陣列
NP	輸入的資料對個數
D	趨近的次方
D2	即為2*D，是參與計算的最大次方和
N	即為D+1，是必須解的聯立方程式個數
A,R,V	解綫性聯立方程式的陣列
P	存放X不同次方的和之陣列
I,J,K,L	廻圈註標
Q,G	工作變數
M	Y的平均值
T	趨近度百分比
XV	欲求出對應Y值的X值
YV	對應特定X值的Y值

研究建議

1. 目前程式中並沒提供資料對的修改。若能在得到一些結果後能再加入或刪除資料的話，就相當的便利。請將此功能加入程式內。
2. 你也許需要別種形式的輸出。爲了順應不同的應用，可印出含實際X值，Y值，計算出來的Y值，以及Y值的誤差情形。
3. 有時已確知某些點（或某些範圍的點）比其他點準確。因此你也許想要在這些點加上比重，更趨近正確。可修改此最小平方法使能允許每個資料對附上一個比重值。請研究此一技術，將其加入程式內。

   （註：你也可將那些比重大的點重覆輸入兩次或三次。但此項方法有一個危險，就是多項式的D值必須小於唯一資料點的個數，否則你會發生零出現於分母的錯誤了。
4. 通常人們都想連續試較高次方的多項式，直到獲一個最理想的趨近度。修改程式使能由操作者輸入其可接受的趨近度，然後程式自動嚐試不同次方的多項式趨近，找出滿足此可接受趨近度的最低次方多項式。

微分方程式(DIFFQN)

解微分方程式

目 的

微分方程式將一個變數的改變量表示爲另一變數的函數。幾乎所有的物理科學都具有此類的關係。因此在很多實際工程問題上常必須解微分方程式。

對於很多微分方程，可找到一個正確的分析式解，然而對許多微分方程，並沒有如此簡單的解答存在。此時微分方程就必須使用電腦程式以數值的方式來解。

微分方程的類型極多，此程式專解最簡單的一種，即爲一次微分方程式。此類方程式可寫成以下的型式：

$$\frac{dY}{dX} = (\text{ X與Y的函數 })$$

此處，X爲自變數，Y爲應變數。此方程式即爲Y對X的微分。式子右邊爲含X及Y的運算式。

欲使用此程式，你必須輸入待解的微分方程式。其步驟在“使用方法”內說明。此程式使用一種技術，稱爲四次藍古塔法（fourth-order Runge-Kutta），來解微分方程式。由於篇幅的限制，就不在此詳細介紹，你可參考書目所列的數值方法書籍。此程式允許兩種形態的輸出，你可使用列表顯示或圖形顯示。

使用方法

你必須做的第一件事就是輸入微分方程於程式內。此方程式必須放在200行內。目前此行含一個GOTO指令。若你沒修改200以前先執行此程式，此行的GOTO指令將會產生一個錯誤訊息。200行的格式應該爲：

200 D =（你的X，Y函數）

D代表dY / dX。會對200行做一些GOSUB以設X與Y目前的值。因此，當每次做RETURN時，即能對此特定的X值與Y值，設好其對應的dY / dX值到D內。若需要的話，你可使用200行到899行之間來完成D的定義。899行永遠含一個RETURN指令，因此你不需另外加入。

程式首先提醒你必須先輸入方程式於程式200行內。你確定已做好此項工作，則按入C鍵以繼續程式的進行。接著就必須輸入各種起始狀態，程式催促你

一次輸入一個。這些資料包括：X與Y的起始值，輸出顯示的X間隔以及X的終結值。

現在你可選擇兩種輸出的其中一類。若按入T鍵，則爲表格輸出；或按入G爲圖形輸出。表格輸出爲兩列值，其分別爲X及對應的Y值。圖形輸出以Y軸爲水平軸，X軸則由螢幕垂直下來。此種圖形輸出還要求你設定螢幕顯示的Y軸上限及下限。若你選定圖形輸出，程式即會催促你輸入此值。以星號（ * ）來畫Y的值。若Y軸超出上下限，則在適當的邊上畫出星號。

此輸入狀態完成後，程式開始準備輸出。在螢幕的左下方會顯示一個問號，以通知你程式等你按入任何一鍵就能開始輸出。

輸出乃依X值的間隔一行一行顯示，直到X的終結值。可按入任何一鍵以暫時停止輸出，然後再任按一鍵以恢復輸出。由於你隨時可重新顯示輸出，因此可輕鬆地看這些表格輸出及圖形輸出。

範例問題

問題：有一物體，原本靜止，承受到2000 達因的力。它的原來重量爲20 克，但它移動時，以1克/秒的比率在失去其重量。空氣的阻力恰等於其移動速度的兩倍。描述此類運動的微分方程式爲：

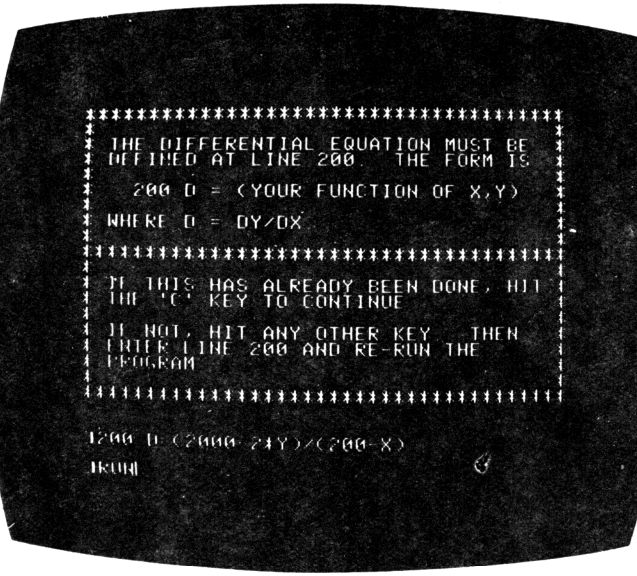
$$\frac{dY}{dX} = \frac{(2000 - 2Y)}{(200 - X)}$$ 其中 Y＝速率（ cm/sec ）

X＝時間（ sec ）

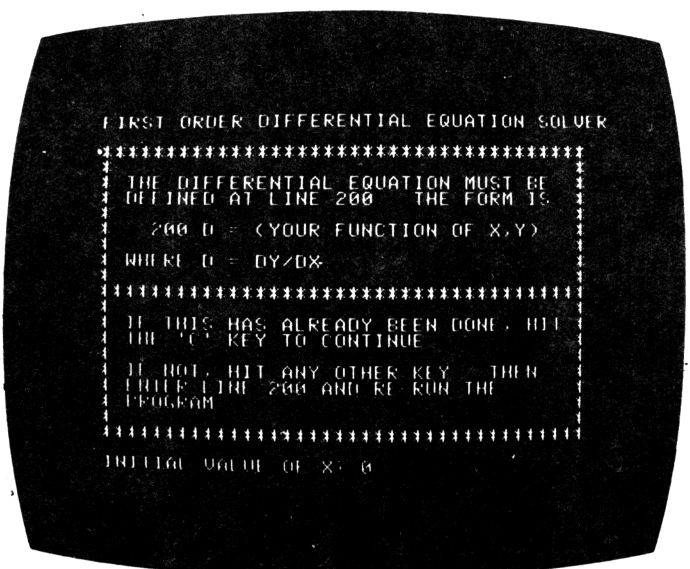
求此物體在前兩分鐘半的速率，以 10 秒爲間隔。而且畫出時間與速率的圖形。

解答：只要輸入此微分方程式，執行此程式即可。

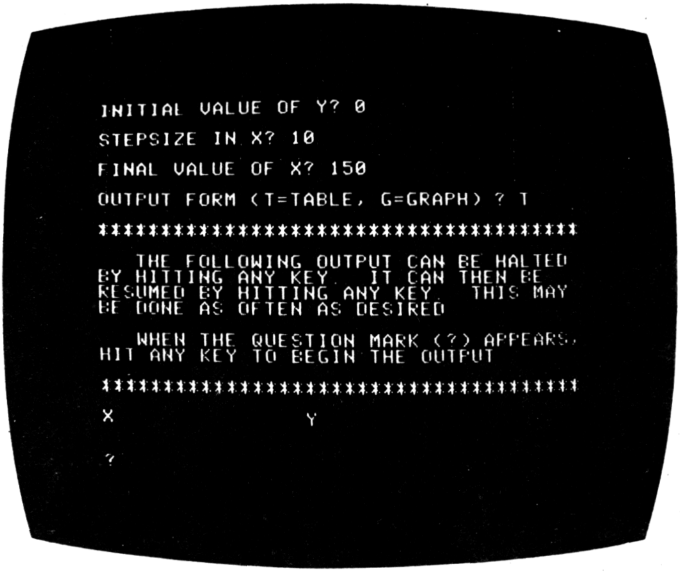
執行範例



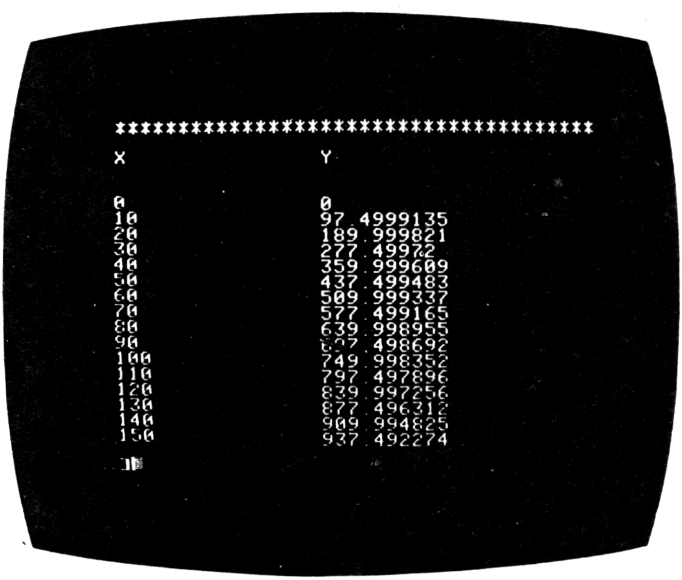
操作者按一鍵出到程式外，輸入方程式於 200 行內。按RUN 以執行程式。



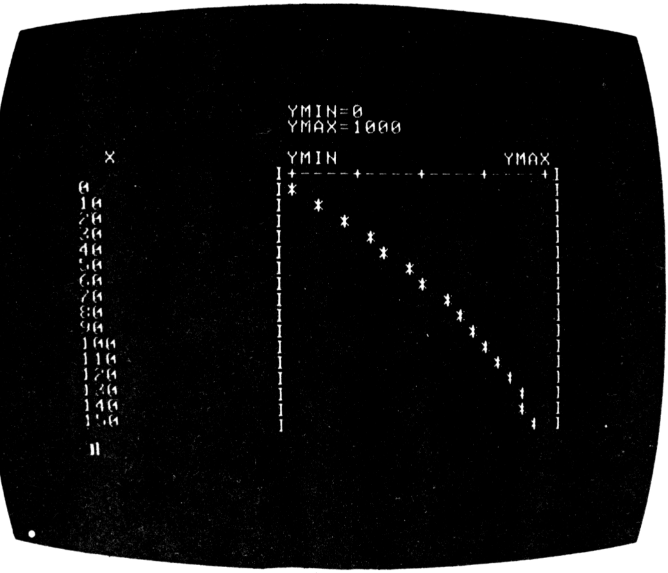
操作者按“ C ”鍵，程式開始進入輸入狀態，操作者回答第一個問題。



操作者完成輸入開始要求列表輸出，程式以問號來回答等操作者按一鍵後，才開始顯示。



操作者按一鍵後，程式顯示列表輸出，X 單位爲 sec Y 爲速度 cm/sec 。



程示顯示圖形輸出。在此之前，程式會詢問Y軸的上下限。其輸入 1000 與 0 ，所以程式顯示他所需圖形。

程 式

```
100 REM: DIFFEQN
110 REM: COPYRIGHT 1980 BY PHIL FELDMAN AND TOM RUGG
130 TEXT
150 GOTO 1200
190 REM
200 GOTO 3000:REM  REDEFINE THIS LINE TO BE D=(YOUR
    FUNCTION OF X,Y)
899 RETURN
900 REM  ************************
910 REM  DEFINE THE DIFFERENTIAL
920 REM  EQUATION BETWEEN LINES
930 REM  200 AND 899
940 REM
950 REM  LINE 200 MUST BE
960 REM  OVERWRITTEN, MAKING IT
970 REM  THE FIRST LINE OF THE
980 REM  EQUATION.
990 REM ************************
```

```
1000 IF F$="T" THEN PRINT XX,YY:GOSUB 1160:RETURN
1010 F=(YY-YL)/(YH-YL):V=INT(17+20*F+0.5)
1020 C=42:IF YY<YL THEN V=17:C=79
1030 IF YY>YH THEN V=37:C=79
1040 PRINT XX;TAB(16);CHR$(73);TAB(V);CHR$(C);TAB(38);
     CHR$(73)
1050 GOSUB 1160:RETURN
1100 PRINT TAB(1);:FOR Q=1 TO 38:PRINT"*";:NEXT:PRINT:
     RETURN
1110 PRINT TAB(1);"*";TAB(38);"*":RETURN
1120 PRINT TAB(38);"*":RETURN
1160 Q=PEEK(-16384):IF Q<128 THEN RETURN
1170 POKE -16368,0
1180 Q=PEEK(-16384):IF Q<128 THEN 1180
1190 POKE -16368,0:RETURN
1200 HOME
1210 PRINT"FIRST ORDER DIFFERENTIAL EQUATION SOLVER"
1220 GOSUB 1100:GOSUB 1110
1230 PRINT TAB(1);"* THE DIFFERENTIAL EQUATION MUST BE";
     :GOSUB 1120
1240 PRINT TAB(1);"* DEFINED AT LINE 200.  THE FORM IS";
     :GOSUB 1120
1250 GOSUB 1110:PRINT TAB(1);
     "*    200 D = (YOUR FUNCTION OF X,Y)";:GOSUB 1120
1260 GOSUB 1110:PRINT TAB(1);"* WHERE D = DY/DX.";:
     GOSUB 1120
1270 GOSUB 1110:GOSUB 1100:GOSUB 1110
1280 PRINT TAB(1);"* IF THIS HAS ALREADY BEEN DONE, HIT"
     ;:GOSUB 1120
1290 PRINT TAB(1);"* THE 'C' KEY TO CONTINUE.";:GOSUB
     1120
1300 GOSUB 1110:PRINT TAB(1);
     "* IF NOT, HIT ANY OTHER KEY.  THEN";:GOSUB 1120
1310 PRINT TAB(1);"* ENTER LINE 200 AND RE-RUN THE";:
     GOSUB 1120
1320 PRINT TAB(1);"* PROGRAM.";:GOSUB 1120
1330 GOSUB 1110:GOSUB 1100
1400 PRINT:GET R$:IF R$<>"C" THEN END
1410 INPUT"INITIAL VALUE OF X? ";XX
1420 PRINT:INPUT"INITIAL VALUE OF Y? ";YY:Y=YY:X=XX:
     GOSUB 200
1430 PRINT:INPUT"STEPSIZE IN X? ";DX
1440 PRINT:INPUT"FINAL VALUE OF X? ";XF
1450 PRINT:INPUT"OUTPUT FORM (T=TABLE, G=GRAPH) ? ";F$
1460 F$=LEFT$(F$,1):IF F$<>"T" AND F$<>"G" THEN 1450
```

```
1470 IF F$="T" THEN 1600
1480 PRINT:INPUT"MINIMUM Y FOR THE GRAPH AXIS? ";YL
1490 PRINT:INPUT"MAXIMUM Y FOR THE GRAPH AXIS? ";YH
1500 IF YH>YL THEN 1600
1510 PRINT
1520 PRINT"*** ERROR! -- MAX Y MUST BE > MIN Y ***"
1530 GOTO 1480
1600 PRINT:GOSUB 1100:PRINT
1610 PRINT"   THE FOLLOWING OUTPUT CAN BE HALTED"
1620 PRINT"BY HITTING ANY KEY.  IT CAN THEN BE"
1630 PRINT"RESUMED BY HITTING ANY KEY.  THIS MAY"
1640 PRINT"BE DONE AS OFTEN AS DESIRED.":PRINT
1650 PRINT"   WHEN THE QUESTION MARK (?) APPEARS,"
1660 PRINT"HIT ANY KEY TO BEGIN THE OUTPUT."
1670 PRINT:GOSUB 1100:PRINT
1700 IF F$="T" THEN PRINT"X","Y":PRINT:GOTO 1800
1720 PRINT TAB(17);"YMIN=";YL:PRINT TAB(17);"YMAX=";YH
1730 PRINT:PRINT"  X";TAB(17);"YMIN";TAB(34);"YMAX"
1740 PRINT TAB(16);
1750 PRINT"I+----+----+----+----+I";
1800 PRINT:PRINT"?";
1810 GET R$
1820 PRINT CHR$(8);CHR$(32);CHR$(8);
1830 POKE -16368,0:GOSUB 1000
1900 Q=XX+DX:IF Q>XF+1.E-5 THEN END
1910 X=XX:Y=YY:GOSUB 200:K0=D:X=XX+DX/2:Y=YY+K0*DX/2
1920 GOSUB 200:K1=D:Y=YY+K1*DX/2:GOSUB 200:K2=D
1930 X=XX+DX:Y=YY+K2*DX:GOSUB 200:K3=D
1940 DY=DX*(K0+2*K1+2*K2+K3)/6
1950 YY=YY+DY:XX=XX+DX:GOSUB 1000
1960 GOTO 1900
3000 PRINT:PRINT"*** ERROR! -- YOU HAVE NOT DEFINED"
3010 PRINT SPC(11);"THE DIFFERENTIAL EQUATION"
3020 PRINT SPC(11);"IN LINE 200."
3030 END
```

修改程式

1. 若你已輸入微分方程式，然後想刪掉簡介顯像，可加入此行：

```
1215 PRINT:GOTO 1410
```

此將立即開始進入輸入狀態。

2. 若你想要使用負間隔。可改 1900 行為：

```
1900 Q=XX+DX:IF Q < XF-1.E-5 THEN END
```

主流程

150	開始執行
200-899	操作者所提供的副程式以定義D
1000-1050	輸出顯像
1100-1120	副程式，編排訊息
1160-1190	副程式，暫停及開始輸出
1200-1330	簡介訊息顯像
1400-1530	操作者輸入
1600-1670	顯示其他訊息
1700-1750	設定輸出顯示
1800-1830	等待操作者按入任意鍵以開始輸出
1900-1960	計算每一步的值
3000-3030	錯誤訊息

主要變數

D	dY/dX 值

X , Y	目前X ,Y 值
XX , YY	前一步的X ,Y 值
DX	X 的間隔值
XF	X 的終結值
F$	輸出標示字串(T =表格,G =圖形)
YL , YH	圖形Y 軸的下限與上限
F	Y 軸每一點之間的差距
V	圖形輸出的Tab 位置
C	圖形輸出的CHR$ 引數
K0 , K1 , K2 , K3	藍古塔(Runge-Kutta)的係數
R$	操作者輸入的字串
Q	工作變數
J	廻圈指標

研究建議

1. 修改程式,使得圖形輸出在表格輸出之後自動出現。在表格狀態下,可設好Y 值的上下限,然後在圖形輸出時,可自動用來做爲畫圖的邊綫位置。
2. dY/dX 爲X 的函數值,通常爲很有用的資料。因此請修改程式以加入此項輸出於圖形輸出上。
3. 計算所附隨的誤差依間隔值的選定而定。在大部分的情形下,都是嘗試不同的間隔值以確定誤差不大。若答案的改變不大,你就可順理成章的斷

言誤差不大。更便利的方法爲，在計算之中，即嘗試各種間隔值，以確使誤差很小。可研究這些方法，然後將其附加於程式內。

4. 此程式可容易地擴充爲解一對聯立的一次微分方程式。此可大大地增加可解決問題種類。研究這些步驟，將其[illegible]於程式內。

圖表(GRAPH)

顯示函數圖表

目　的

一幅圖畫勝過千言萬言？這句話在數學的函數中是絕對正確的。一幅圖表可讓你一眼看清函數的特性與性質。最大值，最小值，曲度，馬鞍點等等性質在圖表內都無所遁形。

GRAPH 圖表程式可產生操作者所需的二維函數圖形。輸入的圖形必須爲此形式：

Y＝（X的函數）

自變數X軸爲水平方向，而應變數Y則爲垂直方向。你還必須控制X軸與Y軸的單位。

使用方法

在執行程式前，你必須先輸入欲繪圖的函數。此函數必須置於由200行起的副程式內。其必須定義Y爲X的函數。程式會以不同的X值來呼叫此副程式。此副程式也許爲一行或多行，依函數的複雜程度而定。999行設爲RETURN 指令，因此你不須加入任何的RETURN。

輸入此副程式後，你就可開始執行此程式。程式首先會打出訊息，提醒你必須將函數輸入200行。然後問你X的範圍(即圖表中X值的上限及下限。)上下限的值必須符合下面兩個條件：上限的值必須大於下限值；且上下限值必須在8個字母之內。

接著你必須選擇Y的顯像單位。若想使Y的顯像單位較理想，你也許必須知道在此X範圍內，Y的最大值與最小值。程式可幫你計算此值，顯示出來讓你參考。然後你必須選定Y軸的上限及下限。只要滿足上面兩個條件的值，程式即能接受。

程式接著要求你按入任意鍵以通知程式開始顯示圖表。每個軸都可容納 20 個分格，其X軸與Y軸的原點分別爲其下限。每個分格就依各軸的上限下限單位適當的劃分。而特別顯示下限、中點、以及上限的值(有時候程式沒顯示各軸中點的值，但至少會顯示下限及上限)。

實際繪出的函數曲綫依各軸分格的八倍解析度

來劃。亦即各畫了160個X值及Y值。此種繪圖能力使用了Apple的高解析度圖形。

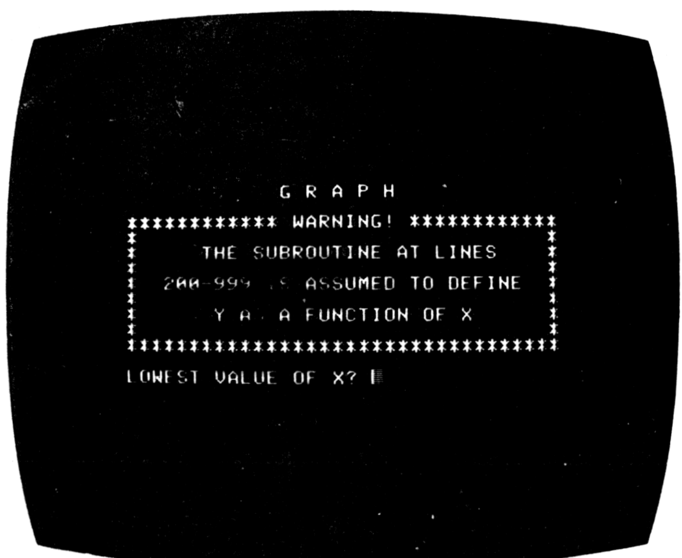
若Y的值超出Y軸的範圍，則會在適當的X值上畫上特殊的粗綫。若Y值太大，則在Y的上限上畫粗綫；若Y值太小，則在Y的下限上畫粗綫。

圖表劃好後，你可按入任何鍵以停止程式執行。

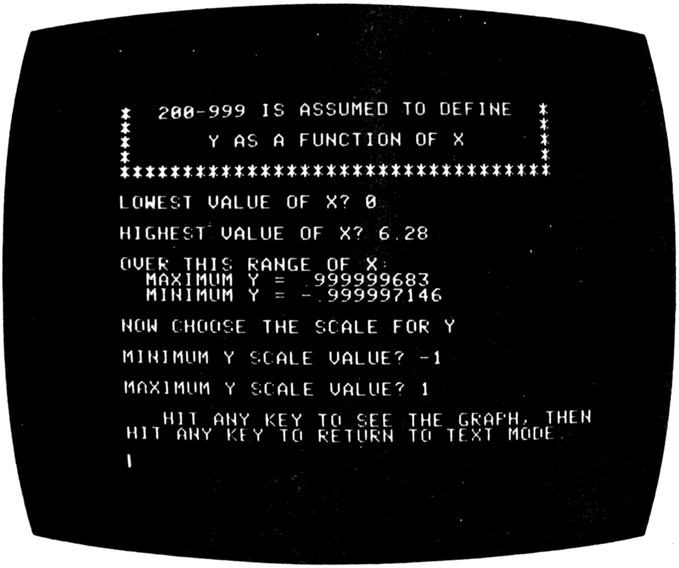
執行範例



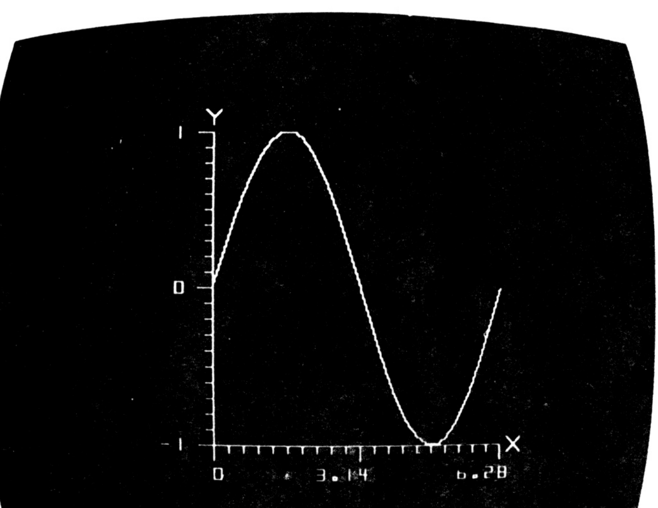
程式搬入後，操作者輸入200行以顯示Y = SIN(X)的圖表。按入RUN　後執行程式。



程式開始，輸入對話。



輸入對話。操作者要求X範圍爲0 － 6.28 程式回答此範圍內的Y值最大及最小值。操作者輸入Y軸的範圍。



圖表顯示。程式等待操作者按一鍵以回到文字狀態。

程　式

```
100 REM: GRAPH
110 REM: COPYRIGHT 1980 BY PHIL FELDMAN AND TOM RUGG
150 GOTO 1200
170 REM
180 REM  ** ENTER SUBROUTINE AT 200 **
190 REM
200 REM  ***** Y=F(X) GOES HERE
999 RETURN
1200 TEXT:HOME:PRINT TAB(13);"G R A P H":PRINT
1210 PRINT TAB(1);:FOR J=1 TO 12:PRINT"*";:NEXT
1220 PRINT" WARNING! ";:FOR J=1 TO 12
1230 PRINT TAB(1);"*";:NEXT:PRINT:GOSUB 1290:PRINT
     TAB(1);"*";TAB(7);
1240 PRINT"THE SUBROUTINE AT LINES";TAB(34);"*":GOSUB
     1290
1250 PRINT TAB(1);"*  200-999 IS ASSUMED TO DEFINE  *":
     GOSUB 1290
1260 PRINT TAB(1);"*";TAB(8);"Y AS A FUNCTION OF X";
     TAB(34);"*":GOSUB 1290
1270 PRINT TAB(1);:FOR J=1 TO 34:PRINT"*";:NEXT
1280 PRINT:GOTO 1300
1290 PRINT TAB(1);"*";TAB(34);"*":RETURN
1300 PRINT:INPUT"LOWEST VALUE OF X? ";XL
1310 XL$=STR$(XL):L=LEN(XL$)
1320 IF L>8 THEN PRINT:PRINT
     " *** TOO MANY DIGITS - PLEASE REDUCE":GOTO 1300
1330 PRINT:INPUT"HIGHEST VALUE OF X? ";XU
1340 IF XU<=XL THEN PRINT:PRINT" *** BAD X RANGE ***":
     GOTO 1300
1350 XU$=STR$(XU):L=LEN(XU$)
1360 IF L>8 THEN PRINT:PRINT
     " *** TOO MANY DIGITS - PLEASE REDUCE":GOTO 1330
1370 XM=(XL+XU)/2:XM$=STR$(XM)
1380 IF LEN(XM$)>9 THEN XM$=" "
1400 DX=(XU-XL)/160:X=XL:GOSUB 200:MN=Y:MX=Y
1410 FOR J=1 TO 160:X=XL+J*DX:GOSUB 200
1420 IF Y>MX THEN MX=Y
1430 IF Y<MN THEN MN=Y
1440 NEXT
1450 PRINT:PRINT"OVER THIS RANGE OF X:"
1460 PRINT"  MAXIMUM Y = ";MX
1470 PRINT"  MINIMUM Y = ";MN:PRINT
```

```
1480 PRINT"NOW CHOOSE THE SCALE FOR Y":PRINT
1490 INPUT"MINIMUM Y SCALE VALUE? ";YL:PRINT
1500 YL$=STR$(YL):L=LEN(YL$)
1510 IF L>8 THEN PRINT
     " *** TOO MANY DIGITS - PLEASE REDUCE":PRINT:GOTO
     1490
1520 INPUT"MAXIMUM Y SCALE VALUE? ";YU:PRINT
1530 IF YU<=YL THEN PRINT" *** BAD Y RANGE ***":GOTO
     1450
1540 YU$=STR$(YU):L=LEN(YU$)
1550 IF L>8 THEN PRINT
     "*** TOO MANY DIGITS - PLEASE REDUCE":PRINT:GOTO
     1520
1560 YM=(YU+YL)/2:YM$=STR$(YM)
1570 IF LEN(YM$)>9 THEN YM$=" "
1600 PRINT"   HIT ANY KEY TO SEE THE GRAPH, THEN"
1610 PRINT"HIT ANY KEY TO RETURN TO TEXT MODE."
1620 PRINT:GET D$
2000 HGR2:HCOLOR=3
2100 HP=107:VP=172
2110 HPLOT HP,VP-160 TO HP,VP:HPLOT TO HP+160,VP
2120 FOR J=VP-160 TO VP STEP 8
2130 HPLOT HP-4,J TO HP,J:NEXT
2140 FOR J=VP-160 TO VP STEP 80
2150 HPLOT HP-8,J TO HP,J:NEXT
2160 FOR J=HP TO HP+160 STEP 8
2170 HPLOT J,VP TO J,VP+4:NEXT
2180 FOR J=HP TO HP+160 STEP 80
2190 HPLOT J,VP TO J,VP+8:NEXT
2200 HPLOT HP+164,VP-4 TO HP+172,VP+4
2210 HPLOT HP+164,VP+4 TO HP+172,VP-4
2220 HPLOT HP-4,VP-172 TO HP,VP-168
2230 HPLOT TO HP+4,VP-172
2240 HPLOT HP,VP-168 TO HP,VP-164
2300 H=HP-20:V=VP-163:A$=YU$:GOSUB 5000
2310 H=HP-20:V=VP-83:A$=YM$:GOSUB 5000
2320 H=HP-20:V=VP-3:A$=YL$:GOSUB 5000
2340 H=HP:V=VP+12:A$=XL$:GOSUB 5000
2350 H=HP+80:V=VP+12:A$=XM$:GOSUB 5000
2360 H=HP+160:V=VP+12:A$=XU$:GOSUB 5000
2400 DX=(XU-XL)/160:DY=(YU-YL)/160
2410 AF=0:X=XL:GOSUB 200
2420 IF Y>YU THEN Y=YU:AF=1
2430 IF Y<YL THEN Y=YL:AF=1
```

```
2440 YY=(Y-YL)/DY:H=HP:V=VP-YY
2450 HPLOT H,V
2460 IF AF=1 THEN GOSUB 3500
2500 FOR J=1 TO 160:X=XL+J*DX:GOSUB 200
2510 IF Y>YU THEN Y=YU:AF=1
2520 IF Y<=YL THEN Y=YL:AF=1
2530 YY=(Y-YL)/DY:H=HP+J:V=VP-YY
2540 HPLOT TO H,V
2550 IF AF=1 THEN GOSUB 3500
2560 NEXT
2570 GET D$:TEXT:END
3000 HPLOT H,V TO H4,V:HPLOT TO H4,V6
3010 HPLOT TO H,V6:HPLOT TO H,V
3020 RETURN
3100 HPLOT H4,V TO H,V:HPLOT TO H,V6
3110 HPLOT TO H4,V6:HPLOT H,V3 TO H4,V3
3120 RETURN
3200 HPLOT H,V3 TO H4,V3:HPLOT H+2,V+1 TO H+2,V+5
3210 RETURN
3300 HPLOT H,V3 TO H4,V3:RETURN
3400 HPLOT H+1,V+4 TO H+3,V+4
3410 HPLOT TO H+3,V6:HPLOT TO H+1,V6
3420 HPLOT TO H+1,V+4:RETURN
3500 HPLOT TO H-1,V-1:HPLOT TO H-1,V+1
3510 HPLOT TO H+1,V+1:HPLOT TO H+1,V-1
3520 HPLOT TO H-1,V-1:HPLOT TO H,V
3530 AF=0:RETURN
4100 HPLOT H+2,V TO H+2,V6
4110 RETURN
4200 HPLOT H,V TO H4,V
4210 HPLOT TO H4,V3:HPLOT TO H,V3
4220 HPLOT TO H,V6:HPLOT TO H4,V6:RETURN
4300 HPLOT H,V TO H4,V
4310 HPLOT TO H4,V6:HPLOT TO H,V6
4320 HPLOT H,V3 TO H4,V3:RETURN
4400 HPLOT H,V TO H,V3:HPLOT TO H4,V3
4410 HPLOT H+3,V TO H+3,V6:RETURN
4500 HPLOT H4,V TO H,V
4510 HPLOT TO H,V3:HPLOT TO H4,V3
4520 HPLOT TO H4,V6:HPLOT TO H,V6:RETURN
4600 HPLOT H,V TO H,V6:HPLOT TO H4,V6
4610 HPLOT TO H4,V3:HPLOT TO H,V3:RETURN
4700 HPLOT H,V TO H4,V:HPLOT TO H4,V+2
4710 HPLOT TO H,V6:RETURN
```

```
4800 HPLOT H,V TO H4,V:HPLOT TO H4,V6
4810 HPLOT TO H,V6:HPLOT TO H,V
4820 HPLOT H,V3 TO H4,V3:RETURN
4900 HPLOT H4,V6 TO H4,V:HPLOT TO H,V
4910 HPLOT TO H,V3:HPLOT TO H4,V3:RETURN
5000 FOR J=LEN(A$) TO 1 STEP -1:H4=H+4:V3=V+3:V6=V+6
5010 D$=MID$(A$,J,1):D=VAL(D$)
5020 IF D=0 THEN 5040
5030 ON D GOSUB 4100,4200,4300,4400,4500,4600,4700,4800,
     4900:GOTO 5090
5040 IF D$="0" THEN GOSUB 3000:GOTO 5090
5050 IF D$="E" THEN GOSUB 3100:GOTO 5090
5060 IF D$="+" THEN GOSUB 3200:GOTO 5090
5070 IF D$="-" THEN GOSUB 3300:GOTO 5090
5080 IF D$="." THEN GOSUB 3400
5090 H=H-8:NEXT:RETURN
```

修改程式

1. 你也許想使程式自動設好Y軸的單位。亦即，要程式將Y值的最大值與最小值設為Y軸的上限及下限。只要插入以下幾行就可獲得此種效果：

```
1443 IF LEN(STR$(MX)) > 8 OR LEN(STR$(MN)) > 8
     THEN 1450
1444 IF MX <= MN THEN 1450
1445 YU=MX:YL=MN:PRINT
1447 YU$=STR$(YU):YL$=STR$(YL):GOTO 1560
```

（註：在一些偶發的情況下，程式也許無法自動幫你設Y軸的單位，此時程式必須能反過來要求你設定Y軸的上下限）。

2. 雖然有程式開始的提醒訊息，也許你仍會忘記輸入函數於200行的副程式。在目前的程式，遇到

此狀況時，將會畫出Y＝0的直綫。若你想製造較戲劇性的效果，可將200行改為：

```
200 Y=1/0
```

此時，若你忘了輸入函數，程式立即在你輸入X軸的上下限之後印出以下的訊息，然後停止執行。

```
?DIVISION BY ZERO ERROR IN 200
```

3. 目前的程式使用了第2頁的高解析度的畫圖能力。欲使用Applesoft的16K ROM系統。或32K磁碟Applesoft 系統）（見附錄1）。可做以下的改變，以使用第1頁的記憶體。

```
2000 HGR:POKE-16302,0:HCOLOR=3
```

主流程

150	開始執行
200-999	操作者提供的函數副程式
1200-1290	顯示提醒訊息
1300-1380	操作者輸入X軸的上下限單位
1400-1570	計算Y值的最大值與最小值。操作者輸入Y軸的上下限
1600-1620	等待操作者按入任意鍵
2000-2240	畫圖表的X，Y軸

2300-2360	標示各軸的方格
2400-2570	畫函數曲綫以及結束程式執行
3000-3420	副程式，畫字母O，E，+，－於兩軸的分格
3500-3530	副程式，畫超出上下限的粗綫
4100-4910	副程式，畫兩軸的數値
5000-5090	副程式，畫單位値

主要變數

XL,XM,XU	X軸的下限，中央值及上限
YL,YM,YU	Y軸的下限，中央値及上限
DX,DY	X及Y的單位差額
X，Y	目前的X，Y値
XL$,XM$,XU$	XL，XM，及XU的字串表示
YL$,YM$,YU$	YL,YM，及YU的字串表示
D$	暫時字串變數
D	D$的數値
H，V	分格單位的水平及垂直位置
HP,VP	軸的原點的水平及垂直位置
XY	分格單位的Y方向支距

AF	Y位置標示（0＝界限內，1＝界限外）
H4,V3, V6	H,V支距

研究建議

1. 顯示計算出發生Y值的最大值及最小值的X位置。
2. 圖表顯示後，允許操作者詢問特定X值所對應的Y值。

積分(INTEGRATE)

決定曲線內的面積

目 的

在大部分的工程及數學的工作上，常必須計算積分。此程式以辛普生法（ Simpson's rule ）來做函數數值上的積分。此法以切割近似的方式來求積分值，直到你對其解的準確度滿意爲止。

若各位學過微積分，應當對此基本的積分原理相當的熟悉。計算X = ℓ （ 下限 ）及X = u （ 上限 ）之間函數的積分就相當於計算其函數曲綫下的面積。註：即圖一的斜綫部分面積。

我們可將這個區域切割成一段一段條形狀，然後以長方形趨近此一條形狀的面積，再將所有長方形面積加起來，即得到近似的面積——積分值。有數學素養的人應該都知道辛普生法可表示爲：

$$\int_{x=\ell}^{x=u} f(x)\,dx \cong \frac{\Delta}{3}\left\{ f(\ell) + f(u) + 4\sum_{j=1}^{N/2} f[\ell + \Delta(2j-1)] + 2\sum_{j=1}^{(N-2)/2} f[\ell + 2\Delta j] \right\}$$

其中N 爲整個區域所切割的數目。在圖一中，N 爲 4。欲參考數值積分方法的書可看：McCracken, Dorn, Numerical Methods and FORTRAN Programming, New York, Wiley, 1964, p.160.不必讓書名的FORTRAN字眼所嚇到。書中的討論跟程式語言無直接關連，只是範例以FORTRAN 所寫的罷了。

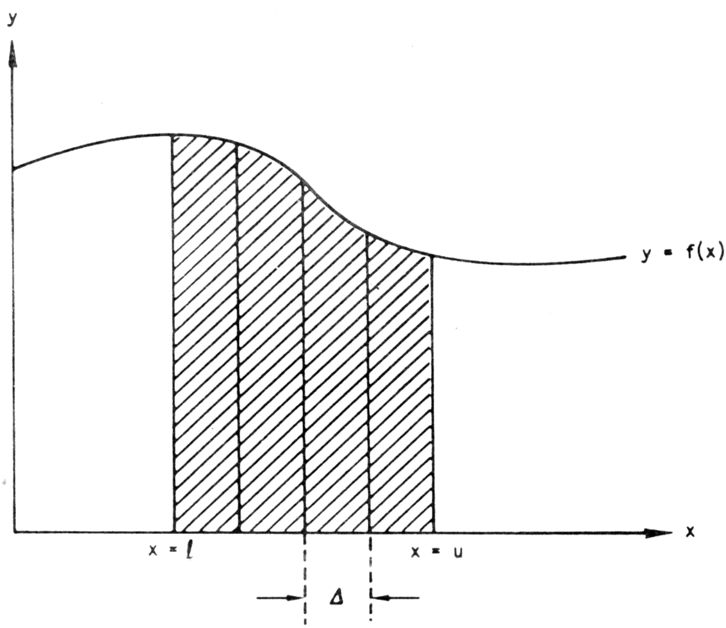


圖 1　f (x)的積分

使用方法

程式剛開始爲一個提醒訊息。其目的爲提醒你必須先將欲積分的函數，以Y爲X的函數形式輸入於200行開始的副程式。

然後要求你提供積分範圍的上限與下限。任何數值都可接受，甚至X值的下限都可不小於上限。

程式即開始顯示計算出來的積分值。每一回合所計算的切割個數以加倍增加。此可讓積分值的正確性依計算時間的增長而增加。對大部分的函數而言，你可看出積分值快速地趨近於常數(或接近常數)。當然，此值即爲最適當的積分值。

當你對解答的正確性相當滿意後，可按入控制C鍵以結束程式的執行。若非如此，程式將會永遠執行下去。而每次下回合的計算過程大約爲前回合的兩倍。亦即下一回合的計算過程所花費的時間大約爲前面所有回合所花的總時間。因此幾回合下來，執行一回合就要花費Apple 電腦幾小時甚至幾天的時間。然後，計算時所發生的捨去誤差就慢慢蓋過收斂的正確性，使常數再度地改變。然而電腦的浮點數計算的準確度會遲延此現象的發生。也許尚未看到此現象時，你已失去自己的耐心了。

欲做積分的函數難易度依你的需要而定。可寫成一行或好幾百行。不管何種情形，此函數的副程式必須由200行開始。在程式中，以各個不同的X值呼叫

此副程式。當返回（Return）時，必須設好此給定的X值所對應的Y值。此副程式必須能夠計算積分範圍內的所有X值。

若你的函數含分散的X實驗數據，那你必須做一些工作使副程式能夠計算分散X值中間的函數值。我們建議以下兩個方式：首先，你可寫一個副程式做Y值的綫性內插。其概念為搜尋你的資料列表，找出上下包圍欲求函數值之X值的一對X值。其二為：使用本節的CURVE曲綫程式以求得近似的多項式趨近你的實驗數據。此種多項式可很容易地輸入200行的副程式內。

順便提及，辛普生法對任何低於三次的多項式而言，其值是正確的。亦即，若函數可寫成以下的形式：

Y = A*X↑3 + B*X↑2 + C*X + D

其中，A，B，C，D為常數。則程式可正確的求得積分值。

執行範例

此處以下列的積分做為執行範例：

$$\int_{x=0}^{x=1} \frac{4}{1+x^2}\,dx$$

此例子理論上的積分值為π。你應很熟悉π圓周率的

値爲3.1415926535 …。在程式執行以前，必須先將上面的函數輸入200行。爲了讓你參考，將計算積分値所經過的時間每行輸出的括號內，第一項爲小時，接著分鐘，接著秒鐘。此可讓你對程式每回合的計算所花時間有個確實概念。但這些時間並不會眞正顯示於程式實際的輸出。

INTEGRATE - SAMPLE RUN

```
]200 Y=4/(1+X*X)

]RUN

          INTEGRAL BY SIMPSON'S RULE


************ WARNING! ************
*                                *
*    THE SUBROUTINE AT LINES     *
*                                *
*  200-999 IS ASSUMED TO DEFINE  *
*                                *
*      Y AS A FUNCTION OF X      *
*                                *
**********************************

LOWER LIMIT OF X? 0

UPPER LIMIT OF X? 1

#SEGMENTS        INTEGRAL

2                3.13333334        (00:00:00)
4                3.14156863        (00:00:00)
8                3.1415925         (00:00:00)
16               3.14159265        (00:00:01)
32               3.14159266        (00:00:02)
64               3.14159265        (00:00:03)
```

```
128                 3.14159266          (00:00:06)
256                 3.14159266          (00:00:12)
512                 3.14159265          (00:00:23)
1024                3.14159266          (00:00:46)
2048                3.14159265          (00:01:32)
4096                3.14159266          (00:03:04)
8192                3.14159267          (00:06.09)
16384               3.14159265          (00:12:21)
32768               3.14159267          (00:24:48)
65536               3.14159265          (00:50:04)

(control C key pressed)
```

程　式

```
100 REM: INTEGRATE
110 REM: COPYRIGHT 1980 BY PHIL FELDMAN AND TOM RUGG
130 TEXT
150 N=2:GOTO 1200
170 REM
180 REM  ** ENTER SUBROUTINE AT 200 **
190 REM
200 REM  ***** Y=F(X) GOES HERE
999 RETURN
1200 HOME:PRINT SPC(4);"INTEGRAL BY SIMPSON'S RULE":
     PRINT:PRINT
1210 PRINT TAB(1);:FOR J=1 TO 12:PRINT"*";:NEXT
1220 PRINT" WARNING! ";:FOR J=1 TO 12
1230 PRINT"*";:NEXT:PRINT:GOSUB 1290:PRINT TAB(1);"*";
     TAB(6);
1240 PRINT"THE SUBROUTINE AT LINES";TAB(34);"*":GOSUB
     1290
1250 PRINT TAB(1);"*  200-999 IS ASSUMED TO DEFINE  *":
     GOSUB 1290
1260 PRINT TAB(1);"*";TAB(8);"Y AS A FUNCTION OF X";
     TAB(34);"*":GOSUB 1290
1270 PRINT TAB(1);:FOR J=1 TO 34:PRINT"*";:NEXT
1280 PRINT:GOTO 1300
1290 PRINT TAB(1);"*";TAB(34);"*":RETURN
1300 PRINT:INPUT"LOWER LIMIT OF X? ";L
```

```
1310 PRINT:INPUT"UPPER LIMIT OF X? ";U
1360 PRINT:PRINT:PRINT
1370 PRINT"# SEGMENTS","INTEGRAL"
1380 PRINT
1400 DX=(U-L)/N:T=0
1410 X=L:GOSUB 200:T=T+Y
1420 X=U:GOSUB 200:T=T+Y
1450 M=N/2:Z=0
1460 FOR J=1 TO M
1470 X=L+DX*(2*J-1):GOSUB 200:Z=Z+Y
1480 NEXT:T=T+4*Z
1500 M=M-1:IF M=0 THEN 1600
1510 Z=0:FOR J=1 TO M
1520 X=L+DX*2*J:GOSUB 200:Z=Z+Y
1530 NEXT:T=T+2*Z
1600 A=DX*T/3
1610 PRINT N,A
1620 N=N*2
1630 GOTO 1400
```

修改程式

1. 你也許想使程式於計算特定的切割數目的積分後，能夠自動的停止。加上如下一行，可讓程式於計算超過 100 個切割的積分後，自動的停止。

   1615 IF N >= 100 THEN END

   當然，你可用別的數取代 100 。

2. 也許你想於計算積分之時看看切割段數的改變的情形。可直接修改 1620 行。欲增加切割段數的改變率，可試試：

   1620 N=N*4

欲將改變率修改爲加上固定的數，可試：

1620 N=N+50

注意，必須確使N的值隨時爲偶數。

主流程

150	開始及呼叫主要程序
200-999	操作者提供的副程式以計算函數 f (x)
1200-1290	顯示簡介訊息及提醒訊息
1300-1310	操作者輸入積分範圍
1360-1380	顯示各欄標題
1400-1420	計算每一點的積分因子
1450-1480	由一合計加入積分因子
1500-1530	由另一合計加入積分因子
1600-1630	完成積分計算及顯示此結果。增加分割段數然後重新計算。

主要變數

N	切割段數
J	廻圈註標
L , U	X 的積分下限與上限
DX	一段的寬度

T	積分的部分結果
M	合計的個數
Z	合計的部分和
A	積分值
X	目前的X 值
Y	目前的函數值 $Y = f(x)$

研究建議

1. 研究數值積分其他類似技術，譬如較簡單的梯形法。然後增添一個輸出欄輸出新方法的積分值。比較此兩種方法如何收斂到正確的答案。
2. 修正程式以雙精確度（ double precision ）來計算積分值。註：至少 15 位準確度。試一下執行範例所使用的函數看看是否你能夠以如此高的準確度算出 π 值。

聯立方程式(SIMEQN)

解線性聯立方程式

目　的

SIMEQN 聯立方程式程式可解一組聯立綫性方程式。此類問題常常發生於工程及數學的工作上。當學生面臨一些以大量文字來敍述的問題時，通常可將其建立成適當的聯立方程式而得以解決。

此程式可處理20個式子有20個未知數的聯立方程式，足以應付所有的實際應用問題。欲解的聯立方程式可用數學式表示爲以下形式：

$$
\begin{array}{l}
A_{11}X_1 + A_{12}X_2 + \ldots + A_{1N}X_N = R_1 \\
A_{21}X_1 + A_{22}X_2 + \ldots + A_{2N}X_N = R_2 \\
\quad\vdots \\
A_{N1}X_1 + A_{N2}X_2 + \ldots + A_{NN}X_N = R_N
\end{array}
$$

N 爲式子的個數，也爲未知數的個數。未知數以 X_1 到 X_N 表示。

每個方程式內的每項未知數都有一個係數，以及右邊的常數項。這些係數（A矩陣）及常數項 (R_1 到 R_N) 都必須為常數——正數，負數或零。A矩陣以雙註標表示，第一個註標為方程式號碼，第二個註標為未知數的號碼。

使用方法

程式會催促你做所有必須的輸入。首先，它詢問共有多少個方程式（亦即多少個未知數）構成你的聯立方程式。此數目至少為1且不大於20。

接著你必須輸入聯立方程式的係數及常數項，程式會要求你一次輸入一個項，指示目前它所需要的項。

一旦你輸入完成後，程式開始計算其聯立解。若N值大的話，必須花費較多的時間。程式顯示結果後，即結束執行。當然，其答案結果為 X_1 到 X_N 這些未知數的值。

若你感興趣的話，程式所使用的解聯立方程式數值方法稱為高斯消去法（Gaussian elimination）。其使用行的代入與消去產生中樞項（即為此行方程式係數最大的項）。然後倒回代換以得到最後的答案。此技術比它的表面說法還簡單，而且很容易寫成程式。可見書目所列數值方法的書。

範例問題

問題：有一畫家持有大量的三種不同的顏料：深綠色，淺綠色，以及純藍色。深綠色內含30％的藍色素，20％的黃色素其餘爲底色。淺綠色內含10％，藍色素35％的黃色素其餘爲底色。純藍色含90％的藍色素無黃色素其餘爲底色。畫家需要一種中等的綠色，內含25％藍色素，25％黃色素。他必須以何種比例來混合手中的三種顏料？

解答：設 X_1 ＝深綠色的使用比例

X_2 ＝淺綠色的使用比例

X_3 ＝純藍色的使用比例

可將問題化爲以下的三個聯立方程式：

$$
\begin{aligned}
0.3\,X_1 + 0.1\,X_2 + 0.9\,X_3 &= 0.25 \\
0.2\,X_1 + 0.35\,X_2 &= 0.25 \\
X_1 + X_2 + X_3 &= 1.0
\end{aligned}
$$

第一個式子代表在混合的顏料中，藍色素的數量。第二個式子代表黃色素的數量。第三個式子表示混合顏料純粹是以此三個顏料混合而成的（注意，所有的百分比都表示爲0到1之間）。此問題可使用SIMEQN程式得到以下的結果。

執行範例

```
A SIMULTANEOUS LINEAR EQUATION SOLVER

HOW MANY EQUATIONS IN THE SET? 3

THE 3  UNKNOWNS WILL BE DENOTED
X1 THROUGH X3
------------------------------------

ENTER THE PARAMETERS FOR EQUATION 1

COEFFICIENT OF X1? .3
COEFFICIENT OF X2? .1
COEFFICIENT OF X3? .9
RIGHT HAND SIDE? .25

------------------------------------

ENTER THE PARAMETERS FOR EQUATION 2

COEFFICIENT OF X1? .2
COEFFICIENT OF X2? .35
COEFFICIENT OF X3? 0
RIGHT HAND SIDE? .25
------------------------------------

ENTER THE PARAMETERS FOR EQUATION 3

COEFFICIENT OF X1? 1
COEFFICIENT OF X2? 1
COEFFICIENT OF X3? 1
RIGHT HAND SIDE? 1

------------------------------------

THE SOLUTION IS

  X1= .55
  X2= .4
  X3= .05
```

因此畫家必須使用55 %深綠，40 %淺綠以及5 %純藍。

程　式

```
100 REM: SIMEQN
110 REM: COPYRIGHT 1980 BY PHIL FELDMAN AND TOM RUGG
130 TEXT
150 M=20
200 HOME
210 PRINT"A SIMULTANEOUS LINEAR EQUATION SOLVER"
220 PRINT:PRINT
300 INPUT"HOW MANY EQUATIONS IN THE SET? ";N
310 PRINT:N=INT(N):IF N>0 AND N<=M THEN 330
320 PRINT"** ERROR! ** IT MUST BE BETWEEN 1-";M:PRINT:
    GOTO 300
330 DIM A(N,N),R(N),V(N)
340 PRINT"THE ";N;" UNKNOWNS WILL BE DENOTED"
350 PRINT"X1 THROUGH X";STR$(N)
360 GOSUB 900:FOR J=1 TO N
370 PRINT"ENTER THE PARAMETERS FOR EQUATION ";J
380 PRINT:FOR K=1 TO N
390 PRINT"COEFFICIENT OF X";STR$(K);
400 INPUT"? ";A(J,K):NEXT
410 INPUT"RIGHT HAND SIDE? ";R(J)
420 GOSUB 900:NEXT
430 GOSUB 2000
500 PRINT"THE SOLUTION IS":PRINT
510 FOR J=1 TO N
520 PRINT"  X";STR$(J);" = ";V(J)
530 NEXT:END
900 PRINT:PRINT"-----------------------------------"
910 PRINT:RETURN
2000 IF N=1 THEN V(1)=R(1)/A(1,1):RETURN
2010 FOR K=1 TO N-1
2020 I=K+1
2030 L=K
2040 IF ABS(A(I,K))>ABS(A(L,K)) THEN L=I
```

```
2050 IF I<N THEN I=I+1:GOTO 2040
2060 IF L=K THEN 2100
2070 FOR J=K TO N:Q=A(K,J):A(K,J)=A(L,J)
2080 A(L,J)=Q:NEXT
2090 Q=R(K):R(K)=R(L):R(L)=Q
2100 I=K+1
2110 Q=A(I,K)/A(K,K):A(I,K)=0
2120 FOR J=K+1 TO N:A(I,J)=A(I,J)-Q*A(K,J):NEXT
2130 R(I)=R(I)-Q*R(K):IF I<N THEN I=I+1:GOTO 2110
2140 NEXT
2150 V(N)=R(N)/A(N,N):FOR I=N-1 TO 1 STEP -1
2160 Q=0:FOR J=I+1 TO N:Q=Q+A(I,J)*V(J)
2170 V(I)=(R(I)-Q)/A(I,I):NEXT:NEXT
2180 RETURN
```

修改程式

1. 程式目前可容納最多 20 個方程式。若你需要更多，而且你的系統具有足夠的記憶空間。則可修改 150 行的M 值以擴大能使用的方程式個數。例如，欲容納 50 個方程式，50 個未知數，可修改 150 行爲：

   150 M=50

2. 有時你也許會驚訝地發現程式執行失敗而顯示以下的訊息：

   ?DIVISION BY ZERO ERROR IN 2150

   比表示你輸入的係數（A 陣列）形態不良，無法找到正確的答案。有數種因素可發生此現象。譬

如，一個方程式恰爲另一個式子的倍數；或者，特殊未知數的所有係數都爲0。若你希望程式能印出診斷訊息，可加入此行：

```
2145 IF A(N,N)=0 THEN PRINT
     "ILL-CONDITIONED INPUT":STOP
```

主流程

200-220	清除螢幕，顯示標題
300-430	操作者輸入係數及計算答案
500-530	顯示答案結果
900-910	副程式，以控制輸出的格式
2000-2180	副程式，計算聯立解答案。其包括以下部分
2000	若N = 1 可直接得到答案
2010-2140	高斯消去法
2030-2100	交換行以獲得中樞項
2150-2180	倒回代換

主要變數

I,J,K,L	迴圈指標及註標
N	方程式的個數（同時也爲未知數的個數）
A	係數二維陣列

R	常數項陣列
V	聯立解的陣列
Q	工作變數
M	程式可容納的最大方程式數目

研究建議

1. 程式在計算聯立解的時候已修改了A陣列與B陣列。此表示程式無法於顯示結果中包括原始的輸入聯立方程式。修改此程式使能儲存原來的聯立方程式，使能顯示於結果內。
2. 目前的程式無法更改輸入的錯誤。請修改程式使能做輸入資料的修改。

統計(STATS)

計算各統計量

目　的

您是否曾經想過自己只是一些統計資料？也許您會猛然發覺自己只是電腦裡面各種資料的一部分，而感慨不已。您的身高、體重、年齡、學歷、職業、薪水、存摺等等，在電腦內也許都已屬於統計資料的一部分。因此也許不少人對“統計”一詞有一種排斥感，認爲其爲無人性、騙人的玩意。但不可否認地，統計的概念已深入人心，與我們的生活息息相關。當我們說“他爲中等身材”或者“今年最熱的一天”，就已經將統計的概念注入我們的日常生活中了。

當然，若使用得當，統計是一個相當有用的分析工具。STATS 統計程式可分析您提供的一組數據資料。可編排您的數據，將其依大小分類，然後計算一些統計的參數以描述這些數據的特性。

這些工作可應用極廣。教師也許想分析一下考試

成績以評定學生的程度。商人也許想統計顧客購物的情形以決定市場趨勢。棒球隊教練想要知道他手下的球員的打擊率及失誤率。您也可以想出一些新的應用。

使用方法

在輸入資料以前，程式首先問您是否要對輸入的資料值使用識別字。識別字可用來辨別各個資料。譬如，以姓名來辨別各人的考試成績，以城鎮和人口相結合，公司名稱與銷售量相結合等等。按入Y或N鍵以回答是否使用辨別字，不須按 RETURN 鍵。

接著您必須輸入資料列表。程式會以問號催促輸入每一個資料值。若使用識別字，則必須在一個資料值前，先輸入其識別字。您可使用任何長度的字串，做爲識別字。然而，若您將它們限制在 18 個字母以內，則以後的輸出格式將會美觀多了。

在此資料輸入狀態的任何時刻都可使用兩個特別的輸入：*END與*BACK。不管是否使用了識別字，都可以使用這兩個字。欲通知程式資料已經輸入完成，可在問號出現時按入此四個字母 *END 。

若您發現資料輸入錯誤，可使用此五個字母字串 *BACK ，使其回到輸入狀態。程式即讓您修改前一項的資料。連續使用 *BACK 即能跳回到前面任何位置。

輸入完成後，程式即進入命令狀態。您有下面四

種選擇命令以繼續程式的執行：

1）以輸入的順序列出資料

2）以分類大小順序列出資料

3）顯示統計參數

4）結束程式

只要按入1，2，3，4鍵即能表示您的選擇。若選定前三個選擇，程式執行選定的工作後，會回到此命令狀態以便操作者做別的選擇。若您選4，則結束程式的執行。以下分別述說各個選擇的細節。

選擇1與2可提供資料列表的印出。選擇1以輸入的順序印出資料列表，而選擇2則將資料由大到小分類。在這兩個選擇，若使用識別字，會跟資料印在一起。

在這兩個選擇中，按入任意鍵即開始印資料列表，列表顯示之中，按入任意鍵即可暫時停止顯示，可允許您慢慢地看資料列表。再按入任意鍵又可恢復顯示。這種顯像的開始與暫停可依需要而任意重覆。當顯示完成後，您必須按入一鍵以重新進入命令狀態。

選擇了可計算資料的統計分析。程式提供了多種統計參數的計算與顯示。以下為各種統計參數的說明。

程式提供有三種位置向心量。這些量指示了資料的平均狀況。期望值（mean，或稱平均值）為資料的總和除以資料的個數。若資料由大至小排好，則中央值（medium）為最中央的資料值，若資料的個數為偶數，則中央值為中間兩個數的平均。半徑值（midrange）為最大數與最小數的差距之一半。

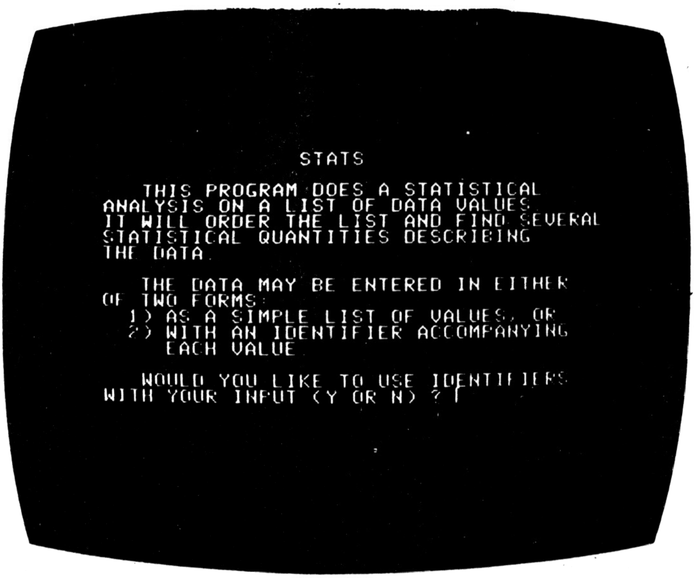
這些資料的位置向心量雖然可說明資料的平均值狀況，却無法表現出資料的分佈狀況。因此我們必須測量其分散狀況，通常稱爲測量變異（measure of variation）。最簡單的變異數爲範圍（range），其爲資料的最大值與最小值之差。另外兩個相關的分散量爲變異數（variance）以及標準方差（standard deviation）。變異數定義爲：

$$VA = \frac{\sum_{i=1}^{N} (V_i - M)^2}{N - 1}$$

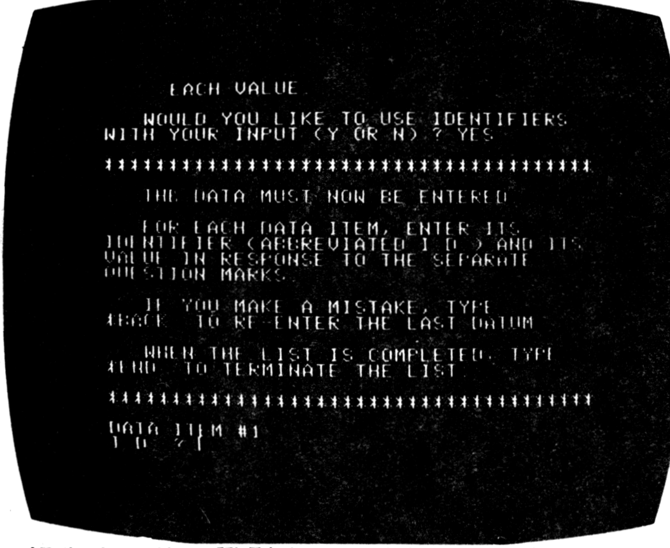
其中N爲資料值的個數，V_i 爲資料 i，而M爲期望值。而標準方差則爲變異數的平方根。由於篇幅的關係，不仔細介紹它們的理論上用途，各位可參考書目所列書籍。其基本原則爲標準方差的值越小，則資料越集中於平均值位置。

尚須提醒一下，當首次選擇 2 或 3 時，程式必須花費一些時間做資料的分類。其所需時間依資料的個數以及資料順序的雜亂程度而定。 25 個資料平均約需 15 秒。 50 個資料約需 1 分鐘。 100 項資料約需 4 分鐘。此時 Apple 電腦正埋頭苦幹地處理，不要誤以爲它在做白日夢或打瞌睡！若您的資料個數極多，可趁此機會開開冰箱，打打電話。

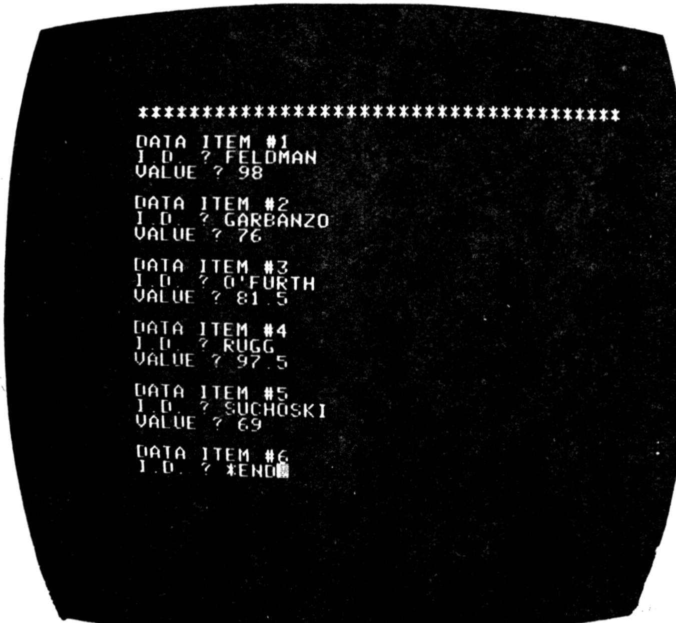
執行範例



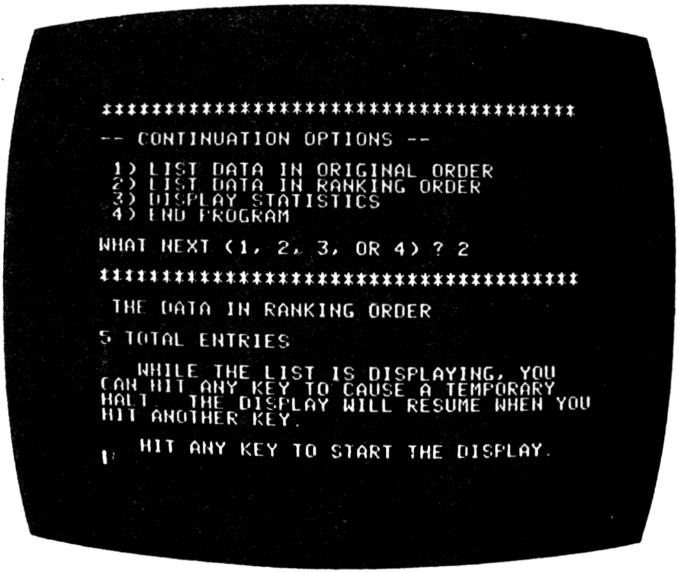
程式顯示簡介說明，詢問操作者是否要使用識別字。



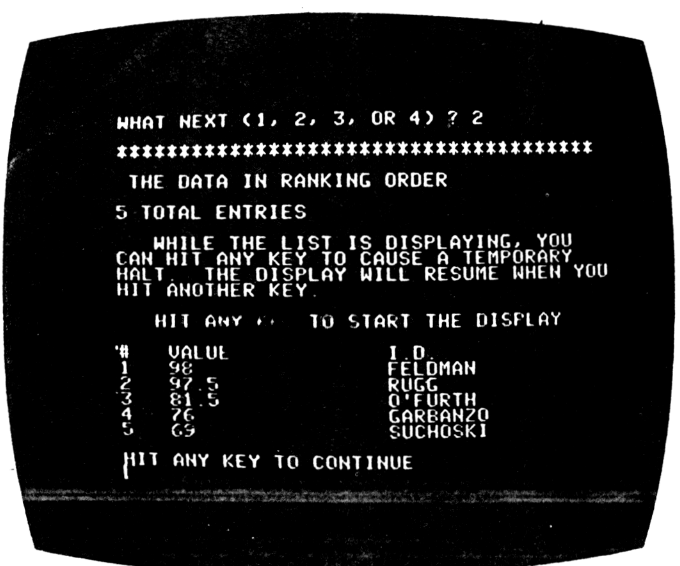
操作者要使用識別字。程式解釋資料輸入方式，開始等待操作者輸入。



操作者輸入性向測驗的成績之後，以 *END 來標示資料結束。



操作者要求列表依序分類，程式等待按一鍵以繼續執行。



操作者按一鍵後，程式顯示了依序分類的列表。然後等待按鍵以繼續。



執行不久後，操作者選擇 3 ，程式即計算顯示各種統計參數。

程 式

```
100 REM: STATS
110 REM: COPYRIGHT 1980 BY PHIL FELDMAN AND TOM RUGG
150 B$="*BACK":E$="*END"
160 MX=100
170 DIM D$(MX),V(MX),Z(MX)
180 Z(0)=0:N$=CHR$(32)
200 TEXT:HOME:PRINT:PRINT
210 PRINT TAB(16);"STATS":PRINT
220 PRINT"   THIS PROGRAM DOES A STATISTICAL"
230 PRINT"ANALYSIS ON A LIST OF DATA VALUES."
240 PRINT"IT WILL ORDER THE LIST AND FIND SEVERAL"
250 PRINT"STATISTICAL QUANTITIES DESCRIBING"
260 PRINT"THE DATA.":PRINT
```

```
270 PRINT"   THE DATA MAY BE ENTERED IN EITHER"
280 PRINT"OF TWO FORMS:"
290 PRINT"  1) AS A SIMPLE LIST OF VALUES, OR"
300 PRINT"  2) WITH AN IDENTIFIER ACCOMPANYING"
310 PRINT"     EACH VALUE.":PRINT
320 PRINT"   WOULD YOU LIKE TO USE IDENTIFIERS"
330 PRINT"WITH YOUR INPUT (Y OR N) ? ";
340 GET R$
350 IF R$="Y" THEN PRINT"YES":F=1:GOTO 400
360 IF R$="N" THEN PRINT"NO":F=0:GOTO 400
370 GOTO 340
400 GOSUB 2100:PRINT
410 PRINT"   THE DATA MUST NOW BE ENTERED."
420 PRINT:IF F=1 THEN 460
430 PRINT"   ENTER EACH VALUE SEPARATELY IN"
440 PRINT"RESPONSE TO THE QUESTION MARK."
450 GOSUB 2000:GOTO 500
460 PRINT"   FOR EACH DATA ITEM, ENTER ITS"
470 PRINT"IDENTIFIER (ABBREVIATED I.D.) AND ITS"
480 PRINT"VALUE IN RESPONSE TO THE SEPARATE"
490 PRINT"QUESTION MARKS.":GOSUB 2000
500 GOSUB 2100:POKE -16368,0:N=1
510 IF N<1 THEN N=1
520 PRINT:PRINT"DATA ITEM #";N
530 IF F=0 THEN D$(N)=N$:GOTO 570
540 INPUT"I.D. ? ";R$:IF R$=E$ THEN 700
550 IF R$=B$ THEN N=N-1:GOTO 510
560 D$(N)=R$
570 INPUT"VALUE ? ";R$:IF R$=E$ THEN 700
580 IF R$=B$ AND F=1 THEN 520
590 IF R$=B$ THEN N=N-1:GOTO 510
600 V(N)=VAL(R$)
610 IF N=MX THEN PRINT:PRINT
    "** NO MORE DATA ALLOWED! **":N=N+1:GOTO 700
620 N=N+1:GOTO 510
700 N=N-1:IF N=0 THEN PRINT
710 IF N=0 THEN PRINT"** NO DATA -- RUN ABORTED **":END
720 GOSUB 2100
730 PRINT:PRINT"-- CONTINUATION OPTIONS --":PRINT
740 PRINT" 1) LIST DATA IN ORIGINAL ORDER"
750 PRINT" 2) LIST DATA IN RANKING ORDER"
760 PRINT" 3) DISPLAY STATISTICS"
770 PRINT" 4) END PROGRAM"
780 PRINT:INPUT"WHAT NEXT (1, 2, 3, OR 4) ? ";R
```

```
790 R=INT(R):IF R<1 OR R>4 THEN 730
800 IF R=4 THEN END
810 ON R GOSUB 1000,1200,1500
820 GOTO 720
1000 GOSUB 2100:PRINT
1010 PRINT" THE ORIGINAL DATA ORDER":PRINT
1020 PRINT N;" TOTAL ENTRIES":GOSUB 2300
1030 PRINT:PRINT TAB(1);"#":TAB(5);"VALUE";
1040 IF F=0 THEN PRINT
1050 IF F=1 THEN PRINT TAB(22);"I.D"
1060 FOR J=1 TO N
1070 PRINT TAB(1);J;TAB(5);V(J);TAB(22);D$(J)
1080 GOSUB 2500
1090 NEXT:GOSUB 2900:RETURN
1200 GOSUB 2100:PRINT
1210 PRINT" THE DATA IN RANKING ORDER":PRINT
1220 PRINT N;" TOTAL ENTRIES"
1230 GOSUB 2700
1280 GOSUB 2300:PRINT:PRINT TAB(1);"#";TAB(5);"VALUE";
1290 IF F=0 THEN PRINT
1300 IF F=1 THEN PRINT TAB(22);"I.D."
1310 FOR J=1 TO N
1320 PRINT TAB(1);J;TAB(5);V(Z(J));TAB(22);D$(Z(J))
1330 GOSUB 2500
1340 NEXT:GOSUB 2900:RETURN
1500 GOSUB 2100:PRINT
1510 PRINT TAB(6);"STATISTICAL ANALYSIS":PRINT
1520 PRINT"YOUR LIST HAS ";N;" VALUES"
1530 NP=0:NN=0:NZ=0:SQ=0:W=0
1540 FOR J=1 TO N:W=W+V(J):SQ=SQ+V(J)*V(J)
1550 IF V(J)>0 THEN NP=NP+1
1560 IF V(J)<0 THEN NN=NN+1
1570 IF V(J)=0 THEN NZ=NZ+1
1590 NEXT:M=W/N:VA=0:IF N=1 THEN 1610
1600 VA=(SQ-N*M*M)/(N-1)
1610 SD=SQR(VA)
1620 PRINT NP;" POSITIVE; ";NN;" NEGATIVE; ";NZ;" ZERO"
     PRINT
1630 GOSUB 2700:PRINT"MINIMUM VALUE = ";V(Z(N))
1640 PRINT"MAXIMUM VALUE = ";V(Z(1))
1650 PRINT"RANGE = ";V(Z(1))-V(Z(N))
1660 PRINT"SUM OF THE VALUES = ";W:PRINT
1670 PRINT"MEAN = ";M
1680 Q=INT(N/2)+1:MD=V(Z(Q)):IF N/2>INT(N/2) THEN 1700
```

```
1690 MD=(V(Z(Q))+V(Z(Q-1)))/2
1700 PRINT"MEDIAN = ";MD
1710 PRINT"MID-RANGE = ";(V(Z(1))+V(Z(N)))/2
1720 PRINT:PRINT"STD. DEVIATION = ";SD
1730 PRINT"VARIANCE = ";VA
1740 GOSUB 2900:RETURN
2000 PRINT:PRINT"   IF YOU MAKE A MISTAKE, TYPE"
2010 PRINT B$;"  TO RE-ENTER THE LAST DATUM."
2020 PRINT:PRINT"   WHEN THE LIST IS COMPLETED, TYPE"
2030 PRINT E$;"  TO TERMINATE THE LIST.":RETURN
2100 PRINT:FOR J=1 TO 38:PRINT"*";:NEXT
2110 PRINT:RETURN
2300 PRINT:PRINT"   WHILE THE LIST IS DISPLAYING, YOU"
2310 PRINT"CAN HIT ANY KEY TO CAUSE A TEMPORARY"
2320 PRINT"HALT.  THE DISPLAY WILL RESUME WHEN YOU"
2330 PRINT"HIT ANOTHER KEY."
2340 PRINT:PRINT"   HIT ANY KEY TO START THE DISPLAY."
2350 GET R$:POKE -16368,0
2360 RETURN
2500 Q=PEEK(-16384):IF Q<128 THEN RETURN
2510 POKE -16368,0
2520 Q=PEEK(-16384):IF Q<128 THEN 2520
2530 POKE -16368,0:RETURN
2700 IF Z(0)=1 THEN RETURN
2710 FOR J=1 TO N:Z(J)=J:NEXT:IF N=1 THEN RETURN
2720 NM=N-1:FOR K=1 TO N:FOR J=1 TO NM:N1=Z(J)
2730 N2=Z(J+1):IF V(N1)>V(N2) THEN 2750
2740 Z(J+1)=N1:Z(J)=N2
2750 NEXT:NEXT:Z(0)=1:RETURN
2900 PRINT:PRINT"HIT ANY KEY TO CONTINUE"
2910 GET R$
2920 RETURN
```

修改程式

1. 目前程式陣列空間最多只能允許100項資料。而程式所需的記憶空間則依陣列保留最大維數MX，以及是否使用識別字而定。若您的系統至少有

6K byte的可用記憶空間（見附錄1），您可使用到 100 個資料項，每個資料項平均可用 10 個字母的識別字。若不使用識別字，則大概需要 5K byte 的可用空間。若您的應用大於 100 項資料，您就必須依此增加 160 行內的 MX 值。若要容納300 項資料，可做以下修改：

```
160 MX=300
```

當然，您必須具有足夠的 RAM 記憶空間以容納這些（若無識別字，大約需 7K byte 可用空間；若有平均 10 個識別字，則需10K byte 可用空間）。若系統只有 4K byte 記憶空間，則程式必須限定於 20 項資料，且無識別字。欲如此，可用：

```
160 MX=20
```

2. 由於可能和您的列表的識別字相衝突，因此您也許想更改結束資料輸入以及跳回前一個輸入資料的特殊字。此兩個特殊字在程式中分別以變數 E$ 及 B$ 加以控制，在 150 行設定其值。若您想用 /DONE/ 來結束資料輸入，以 /LAST/ 跳回前一個輸入。則 150 行可改為：

```
150 B$="/LAST/":E$="/DONE/"
```

3. 若您想讓列表由小到大分類，而非目前的由大到小，可改變程式 2730 行，將大於（＞）改為小

於（<），因此：

```
2730 N2=Z(J+1):IF V(N1)<V(N2) THEN 2750
```

若只做此項修正，則將使統計計算發生一些荒謬的事。譬如最小值將會顯示在最大值的位置，或者資料範圍將莫名其妙地跑出負號。欲治療這些衝突，可再做以下的修正：

```
1630 GOSUB 2700:PRINT"MINIMUM VALUE=";V(Z(1))
1640 PRINT"MAXIMUM VALUE=";V(Z(N))
1650 PRINT"RANGE=";V(Z(N))-V(Z(1))
```

主流程

150－180	設定變數及陣列維數。
200－370	顯示訊息，決定是否使用識別字。
400－620	檢查操作者輸入的資料。
700－710	檢查輸入的資料至少有1項。
720－820	命令狀態——取得操作者下一個選擇，然後以GOSUB跳到適當的副程式。
1000－1090	副程式，以輸入原來順序印出資料列表。
1200－1340	副程式，以分類順序印出資料列表。
1500－1740	副程式，計算及顯示統計參數。
2000－2360	副程式，顯示各種訊息。
2500－2530	副程式，允許操作者暫時開始及停止列表的顯示。

2700－2750 副程式，分類資料列表。

2900－2920 副程式，檢查是否用戶按一鍵以繼續程式執行。

主要變數

MX	可容納的資料個數
D$(MX)	識別字的字串陣列
V(MX)	資料值陣列
Z(MX)	分類好的資料陣列
N	目前所輸入的資料項個數
F	是否使用識別字的標示（1＝是，0＝否）
B$	跳回前一個輸入資料的標示
E$	結束資料輸入的標示
N$	空白識別字字串
R$	操作者輸入字串
NM	N－1
R	繼續執行的選擇
NP	正數資料的個數
NN	負數資料的個數
NZ	零值資料的個數
W	資料值的和
SQ	資料值的平方和
M	平均值
MD	資料的中央值

VA	變異數
SD	標準方差
J，K	迴圈指標
N1,N2	分類之時，做更換的資料位置
Q	工作變數

研究建議

1. 此程式所使用的分類演算法，只在資料的個數很少的狀況下—— 25 個以內，才有不錯的效率。因為它在分類時，不檢查資料列表是否已分類完成。若您的資料列表往往大於 25 項，也許就應該使用較適合大列表的分類技術。可研讀一些分類技術，將其附加於程式內。為了替您起個頭，可試試以下的修改：

```
2720 Q=0:FOR J=1 TO N-1:N1=Z(J)
2730 N2=Z(J+1):IF V(N1)>=V(N2) THEN 2750
2745 Q=1
2750 NEXT:IF Q=1 THEN 2720
2760 Z(0)=1:RETURN
```

若您的資料列表很短，此程序也許比本來的程序還慢。但若列表很長，即可節省極多的時間。

2. 因為程式內以 INPUT 敍述來輸入識別字。而 BASIC 語言會忽略掉逗號後的字。若您使用引號括在識別字的外面，就能避免此種情形，但有

時您也許會忘掉此做法。因此可修改輸入程序，使用一系列的 GET 敍述，則可以一段段地湊成識別字；而且可允許逗號。請修改適當地程序以處理此事。

3. 尚有很多的統計參數可描述資料的性質。研讀以及加入對您有用的統計參數。有一個概念即將資料分組，亦即依資料的範圍區分爲幾塊區域，算算落入各組的資料個數。

第六部分　什錦程式：

有趣的小程式

簡　介

什錦程式可讓各位發現，一些簡短的程式也可以做相當有趣的事。這些程式都有一點數學味道，由於都都相當的簡短，因此最適合初學 BASIC 語言，剛開始寫程式的人做練習。

蒙地卡羅模擬（Monte Carlo Simulation　）可讓程式引導實驗的進行（其並非牽涉到高賭注的賭博）。PI 圓周率程式展現了計算著名的圓周率 π 的近似方法。

直角三角形程式 PYTHAG 找出整數邊長的直角三角形。在此處使用了一個相當聰明的方法。

當您在教室集會時，是否曾環顧四周，納悶是否有人的生日同一天？BIRTHDAY 生日程式會幫您計算一些有趣的現象。

若您知道一些技巧的話，可用 Apple 電腦做一

些高位數的計算。 POWER 次方程式會幫您計算各種整數的次方。並非使用 Apple 一般的九位數準確度，而是使用可到 250 位數的準確度。

生日(BIRTHDAY)

但願同年同月同日生

目　的

假設您進入一個擠滿人的房間內。這些人之中，生日在同一天的機率有多少？必須有多少人在同一個房間，生日同一天的機率才會大於 50 %？在此我們只討論生日的月日，並非年。

此爲相當容易的問題，甚至不須使用電腦。用電腦幫您計算，事情就變得很簡單。此問題最有趣的地方是正確的答案與大部分人們的初步猜測有一段差距。在揭曉以前，先問您，房間內必須擠進多少人，其生日重覆的機率才會高於 50 %？50人、100人還是 200 人？

使用方法

程式開始執行後，首先顯示兩列標題。第一列的標題爲房間內的人數，由 1 開始顯示。第二列標題爲生日重覆的機率。

若爲一個人，當然其機率爲零。因爲沒有別人可與他同日生。若爲兩個人，其機率很簡單是 1 / 365，程式以浮點數表示。（注意，在此我們假設一年爲 365 天，且每個人在任何一天出生的機率相同。）

若三人在房間內，生日重覆的機率爲多少？並非 2 / 365，而是：

1 －（ 364 / 365 乘上 363 / 365 ）

亦即 1 減掉生日不重覆的機率。

若四人在房間內，其機率爲：

1－（364/365 乘 363/365 乘 362/365）

計算的方式依此類推，房間內增加一人，則多乘上一項。您可能很驚訝地發現，生日重覆的機率一下子就超過了 50 %。

此程式一直計算，直到房間內容納 60 個人。若您只要看生日重覆超過 50 %的人數，則必須自己停止程式執行。

執行範例

```
NO. OF    PROB. OF 2 OR MORE
PEOPLE    WITH SAME BIRTHDAY
1         0
2         2.7397261E-03
3         8.20416585E-03
4         .0163559124
5         .0271355736
6         .0404624834
```

程　式

```
100 REM: BIRTHDAY COINCIDENCE PROBABILITY PROBLEM
110 REM: COPYRIGHT 1980 BY TOM RUGG AND PHIL FELDMAN
120 TEXT:HOME
130 PRINT"NO. OF    PROB. OF 2 OR MORE"
140 PRINT"PEOPLE    WITH SAME BIRTHDAY"
150 Q=1
160 FOR N=1 TO 60
170 PRINT N;TAB(11);1-Q
180 Q=Q*(365-N)/365
190 NEXT N
200 END
```

修改程式

1. 更改 160 行的常數 60 ，以改變計算的人數範圍。例如，改爲100 ，可看出機率如何趨近於 1 ；或者改爲 20 ，可在輸出佔滿整個螢幕前結束程

式。

主流程

120－140	顯示標題。
150	設定Q爲1。
160－190	計算生日不重覆的機率，然後計算顯示生日重覆的機率。

主要變數

N	房間內的人數
Q	生日不重覆的機率

研究建議

修改程式，使能允許閏年，而不硬性以一年365天來計算。

圓周率(PI)

計算π的近似值

目 的

希臘字母π，代表數學史上最著名的常數——圓周率。此常數出現於各種數學領域內，而最膾炙人口的例子爲圓的直徑和圓周的關係——若圓的半徑爲 r，則圓周爲 2πr，圓面積爲 πr^2。

由於π爲無理數，無法正確地以十進位數表示。若爲九個有效位數，則π的值爲 3.14159265。幾世紀以來，人們已經發展出好幾種方法來計算π值。

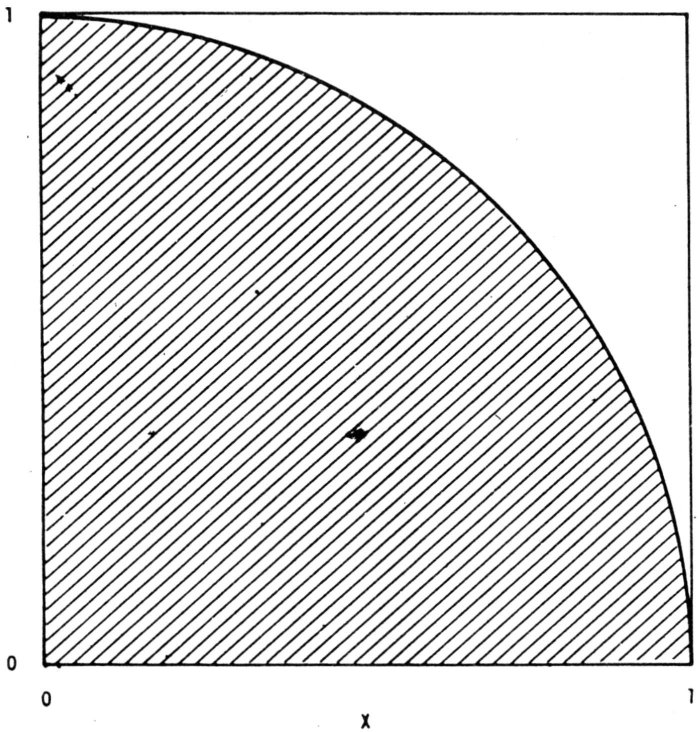
此程式使用一個很有價值的新技術稱爲電腦模擬，來計算π值。"模擬"(simulation)一字本身即能解釋此程式的動作——以電腦幫我們做實驗。這種方式相當受歡迎，有以下幾個原因：在電腦上執行，實驗可以更經濟，無危險性，正確性高，而且有很多實驗在實際世界無法完成。通常電腦有這麼多優點的理由爲其計算速度快，允許模擬可在很短的時間內

做很多次，當然勝過一般曠日費時的實驗了。

此程式模擬擲飛鏢的動作。其飛鏢靶爲特製的，就如圖一所示，是個方形的靶，圓弧所圍的斜線部分是以左下角爲圓心所畫的圓。因此方形的邊長恰爲圓的半徑，其長度均爲 1 。

假設飛鏢靶中任意位置的中鏢機率相同，則飛鏢射中斜線區域的機率就相當於斜線部分的面積除以整個方形靶的面積。

而斜線部分的面積恰爲一個四分之一圓的面積。



圖一　π 飛鏢靶

該記得圓的面積爲 πr^2，其中 r 爲半徑。此時 r = 1，因此整個圓的面積爲 π。所以斜綫部分的面積爲 $\pi/4$。而方形的面積爲 s^2，其中 s 爲一邊的長。而飛鏢靶的一邊長 s ≒ 1，因此整個飛鏢靶的面積爲 1。

現在來算飛鏢射中斜綫部分的機率：

$$射中機率 = \frac{\#射中次數}{\#擲的次數} = \frac{斜綫面積}{靶面積} = \frac{\pi/4}{1} = \frac{\pi}{4}$$

因此我們可用實驗射飛鏢來近似 π 值。我們只要從事飛鏢實驗，計算擊中斜綫部分的比值，然後將此比值乘以 4 即爲所要計算的 π 值。

但並非實際做一個飛鏢靶，手持飛鏢一次一次地擲。我們將擲飛鏢的實驗交給 Apple 電腦來做。程式每次“擲飛鏢”就是在X座標與Y座標分別隨機選定 0 到 1 之間的數。此 X、Y 座標位置就相當於中靶位置。此種隨機數的選定可直接呼叫 BASIC 語言的 RAN 函數。若X，Y中靶位置滿足 $X^2 + Y^2 < 1$，則表示射中斜線部分。

然後程式繼續“擲飛鏢”，計算命中斜綫部分的比率，將其乘以 4 即得 π 的近似值。

使用方法

此程式只要求您做一個輸入，即“每次取樣的個數”，意謂擲多少次後，才開始計算印出目前的結果

。只要大於 1 的任何整數都可以接受。

您輸入取樣個數後，程式即開始著手模擬擲飛鏢，然後顯示其結果。在每次擲滿取樣個數時，即顯示命中的累積總和、以及目前的 π 值近似。

程式一直執行下去，直到您按控制 C 鍵。當您滿意於目前擲飛鏢的次數及近似 π 值後，可按控制 C 鍵以結束程式執行。

執行範例

```
      A DARTBOARD PI CALCULATOR

 SAMPLE SIZE FOR PRINTING? 150


      A DARTBOARD PI CALCULATOR

# HITS      # THROWN        PI
 113         150         3.01333334
 234         300         3.12
 351         450         3.12
 470         600         3.13333334
 587         750         3.13066667
 704         900         3.12888889
 824         1050        3.13904762
 940         1200        3.13333334
 1066        1350        3.15851852
 1185        1500        3.16
 1308        1650        3.17090909
 1425        1800        3.16666667
```

(CONTROL C pressed)

程 式

```
100 REM: PI
110 REM: COPYRIGHT 1980 BY PHIL FELDMAN AND TOM RUGG
160 T=0:TH=0:Q=PEEK(78)+256*PEEK(79):Q=RND(-Q)
300 GOSUB 600
310 INPUT"SAMPLE SIZE FOR PRINTING? ";NP
320 NP=INT(NP):IF NP<1 THEN 300
330 GOSUB 600
340 PRINT"# HITS     # THROWN";TAB(28);"PI"
400 GOSUB 500:TH=TH+NH:T=T+NP:P=4*TH/T
410 PRINT TAB(2);TH;TAB(13);T;TAB(24);P
420 GOTO 400
500 NH=0:FOR J=1 TO NP
510 X=RND(1):Y=RND(1)
520 IF (X*X+Y*Y)<1 THEN NH=NH+1
530 NEXT:RETURN
600 HOME:PRINT TAB(6);
610 PRINT"A DARTBOARD PI CALCULATOR"
620 PRINT:PRINT:RETURN
```

修改程式

1. 若您想讓程式使用固定的取樣個數，可修改310行爲

310 NP=150

當然，此處的值150可依您的喜好而改變。若做此修改後，320行就不再需要，可加以刪除。

2. 若您想讓程式於擲特定次數後自動停止，可加入以下兩行：

```
315 INPUT"TOTAL # DARTS TO THROW";ND
415 IF T>=ND THEN END
```

此項改變會詢問操作者應擲多少次，然後於擲滿之後結束程式執行。

主流程

160	設定常數。
300 − 340	操作者輸入取樣個數，顯示各列標題。
400 − 420	計算及顯示結果。
500 − 530	擲 NP 次飛鏢，記錄命中斜綫部分的次數。
600 − 620	清除螢幕，顯示程式標題。

主要變數

T	擲飛鏢總數
TH	命中總數
NP	取樣個數
NH	在一次取樣中，命中的個數
P	π 值
X，Y	飛鏢擲出的隨機值座標
J	迴圈指標

研究建議

1. 計算程式所計算 π 值的誤差比率，然後與其他結果一起顯示。您必須定義一個新的變數 PI ，設爲 π 的正確值。然後誤差比率爲 PE ，可由以下式子計算出來：

PE = 100*ABS(P-PI)/PI

2. 此程式模擬結果的正確性之主要關鍵爲電腦的隨機數產生器的品質好壞。請研究一下各種隨機數產生器的演算法，然後將其寫入程式內。可更改510行以使用新的演算法。此可實際地用來測試各種隨機數產生器。在書目中的Gruenberger的書，內有各種隨機數產生器的珍貴資料。

乘冪(POWERS)

可到250位數的整數次方

目 的

現在各位大概都很清楚 Apple 電腦內部以九個有效位數來處理數值的運算。若爲小於十億(1,000,000,000)的整數，電腦尚能夠保持正確的值。若對更大的數， Apple 只能記錄最高九個位數(由左算過來)以及其指數部分。亦即，它無法正確處理大於十億的整數的運算，只能以近似的方式完成。對吧？錯了！

此程式即能夠計算整數的階乘(factorial)及乘冪(power)，然後正確地顯示到250個有效位數。程式中使用了“多準確位數的計算”技巧。譬如此程式可告訴您 973 的 47 次方的正確值爲多少。

使用方法

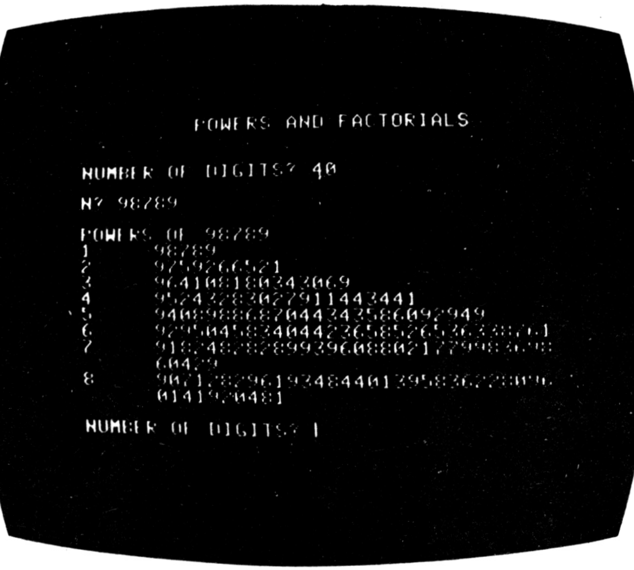
程式首先詢問您最大要以多少位數表示數值，其可爲1到250之間的整數。因此，若您輸入 40 ，則所得的答案最多可到 40 位數。

接著程式要您輸入N的值。若您回答1，則要求程式顯示符合您所聲明的位數的所有階乘。首先，您會看到1的階乘，然後2階乘，依此下去。若您忘了階乘的含意，3階乘爲3乘2乘1其結果爲6；4階層爲4 × 3 × 2 × 1，其結果爲 24 。

若您輸入的N值爲2到100,000之間。則要求程式計算此數遞次的自乘，直到您所聲明的位數範圍。因此若您輸入的N爲 23 ，則以 23 一次方， 23 二次方，然後 23 三次方一直下去。

最後，當程式顯示了符合您的位數之內的最大數值之後，程式再從頭開始。您要求的位數個數越大，程式計算的時間越久。若您輸入零位數，則程式結束。

執行範例



操作者要求計算98789的乘冪到 40 位數。程式計算到98789^8後，詢問下一回合的計算。

程　式

```
100 REM: POWERS AND FACTORIALS
110 REM: COPYRIGHT 1980 BY TOM RUGG AND PHIL FELDMAN
120 TEXT:HOME
130 PRINT TAB(10);"POWERS AND FACTORIALS"
140 PRINT:PRINT
160 DIM N(255)
170 INPUT"NUMBER OF DIGITS? ";M
175 IF M=0 THEN END
180 M=INT(M):IF M>250 OR M<1 THEN 170
190 PRINT:INPUT"N? ";N
200 N=INT(N)
210 IF N<1 OR N>100000 THEN 190
```

```
220 PRINT
230 F=0:IF N=1 THEN F=1:PRINT"FACTORIALS"
240 IF F=0 THEN PRINT"POWERS OF ";N
250 T=10:K=1:N(0)=N
260 FOR J=0 TO M
270 IF N(J)<T THEN 300
280 Q=INT(N(J)/T):W=N(J)-Q*T
290 N(J)=W:N(J+1)=N(J+1)+Q
300 NEXT
310 J=M+1
320 IF N(J)=0 THEN J=J-1:GOTO 320
330 IF J>=M THEN 500
340 D=0:PRINT K;TAB(7);
350 N$=STR$(N(J))
360 D=D+1:IF D>30 THEN D=1:PRINT:PRINT TAB(7);
370 PRINT N$;:J=J-1:IF J>=0 THEN 350
380 N=N+F
390 K=K+1:PRINT
400 FOR J=0 TO M:N(J)=N(J)*N:NEXT
410 GOTO 260
500 FOR J=1 TO 255:N(J)=0:NEXT
510 M=0:N=0:PRINT:GOTO 170
```

修改程式

1. 改寫程式，使其使用固定位數，譬如 50 位數，則刪除 170 行及 180 行，然後插入此行：

170 M=50

2. 欲在輸出開始顯像以前清除螢幕，可改變 220 行爲：

220 HOME

3. 若 250 位數仍不夠您用，可改更大。譬如 500 位

數，可做以下更改：

a. 在160行，改255爲505。

b. 在180行，改250爲500。

c. 在500行，改255爲505。

主流程

120 – 160	顯示標題，設好陣列。
170 – 240	操作者輸入位數個數及N。檢查輸入的合法性，顯示輸出的起頭。
250	設定變數，以便計算。
260 – 300	處理N陣列內的進位，使每個元素內的值不大於9。
310 – 320	回巡N陣列表第一個非零元素。
330	檢查數值是否大於要求的位數。
340 – 370	顯示計數器及數值。若需要的話跳到下一行。
380 – 390	判斷是否該乘N，以取得下一個數值。
400 – 410	N陣列的每個元素都乘上N。跳回260行。
500 – 510	將N陣列歸零以準備下一步的要求，跳到170行。

主要變數

N	計算數值的陣列
M	操作者所要求的有效位數個數
N	起始值。若為1，則為階乘，若大於1，則為N的次方
F	若為階乘則設為0；若為次方，設為1。
T	常數值10
K	目前的次方或階乘的計數器
J	註標變數
Q．W	暫時變數，用來使N陣列的每一元素值在0到9之間
D	目前一行顯示的位數個數（最大30個）
N$	將各個位數轉換成可顯示格式的字串變數。

研究建議

1. 決定可使計算不發生誤差的最大N值（因為中間結果可能超出10億）。然後修改210行，使能允許更大的值進入。
2. 使用類似此程式的技巧，建立一系列的副程式，使能做加法、減法、乘法與除法，或者在兩個陣列間互換數值。然後您就可使用GOSUB指令做高位數的計算。

直角三角形(PYTHAG)

產生直角三角形($a^2+b^2=c^2$)

目　的

該記得畢氏定理吧！直角三角形兩鄰邊的平方和等於斜邊的平方。若以公式表示，則爲 $a^2+b^2=c^2$。符合畢氏定理最讓人熟悉的例子爲3-4-5直角三角形（$3^2+4^2=5^2$）。當然，直角三角形有無限多個。

此程式顯示可構成直角三角形的整數邊a，b，c。

使用方法

使用此程式只要按入 RUN 後，就可以看到直角三角形的三邊長（以 a，b，c 三個邊長構成一個三角形）。程式在每次螢幕上顯示 20 組邊長，然後等待您按入任意鍵（RESET 鍵與E鍵除外），以印下

一個螢幕的 20 組直角三角形。程式繼續算直角三角形，直到您按 E 鍵結束程式執行。

輸出的最左邊一列爲產生的直角三角形計數器。其他三欄分別爲三邊長 a，b，c。

產生的直角三角形的順序並不太明顯，因此在此介紹如何產生此三邊長：

在此處所使用的技術已有人證明出其產生的都爲“基本”直角三角形（其中，“基本”（primitive）表示這些三邊長並非其它直角三角形的三邊長整數倍）。若您有 R 與 S 兩個正整數，則：

1) R 大於 S。

2) R 與 S 爲相反極性（一爲奇數，一爲偶數）。

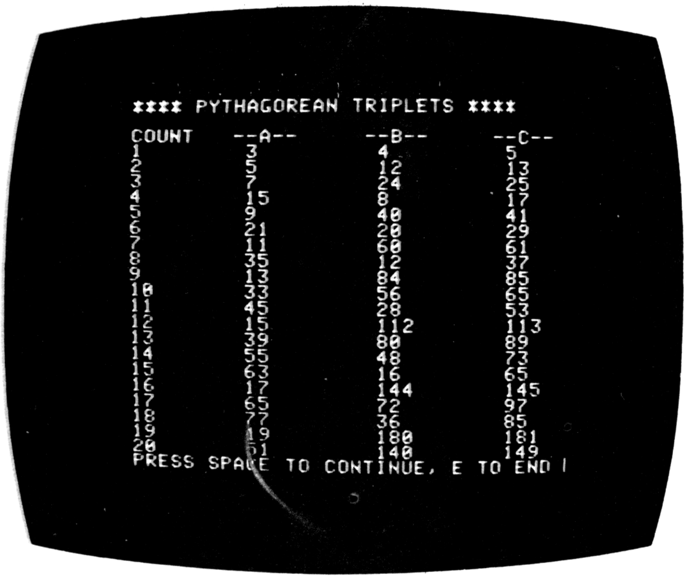
3) R 與 S 互質（他們之間的最大公因數爲 1）。

因此 a，b，c 以下列的方式來計算

$$a = R^2 - S^2$$
$$b = 2RS$$
$$c = R^2 + S^2$$

程式以 R 爲 2 開始計算，產生對此 R 值的所有 S 值（以 R − 1 起，然後遞減），然後 R 再加 1，重覆前面步驟。因此，第一組三邊長就在 R 爲 2 而 S 爲 1 的情形下建立。第二組三邊長則在 R 爲 3 而 S 爲 2 的情況下產生……。

執行範例



程式產生一螢幕的直角三角形，然後等待操作者按鍵以繼續。

程　式

```
100 REM: PYTHAGOREAN TRIPLETS
110 REM: COPYRIGHT 1980 BY TOM RUGG AND PHIL FELDMAN
130 R=2:K=1:D=0
150 GOSUB 350
180 S=R-1
190 A=R*R-S*S
200 B=2*R*S
```

```
210 C=R*R+S*S
220 PRINT K;TAB(10);A;TAB(20);B;TAB(30);C
230 K=K+1:D=D+1:GOTO 400
240 S=S-2:IF S<=0 THEN R=R+1:GOTO 180
250 S1=S:B1=R
260 N=INT(B1/S1)
270 R1=B1-(S1*N)
280 IF R1<>0 THEN B1=S1:S1=R1:GOTO 260
300 IF S1<>1 THEN 240
320 GOTO 190
350 TEXT:HOME
360 PRINT"**** PYTHAGOREAN TRIPLETS ****"
370 PRINT
380 PRINT"COUNT";TAB(9);"--A--";TAB(19);"--B--";TAB(29);
    "--C--"
390 RETURN
400 IF D<20 THEN 240
420 PRINT"PRESS SPACE TO CONTINUE, E TO END ";
430 GET R$
440 IF R$="E" THEN END
450 GOSUB 350:D=0
460 GOTO 240
```

修改程式

1. 改變在130行R的起始值。不要從2開始，試試50或100。
2. 您可以改變螢幕一次顯示的直角三角形個數。只要改變400行的值20即可。也許您不想讓各欄標題捲到螢幕外，因此可將此值改小一點。
3. 欲使程式繼續執行，不須您按一鍵以換下一個螢幕，可插入此行：

   405 GOTO 450

   或

   405 GOTO 460

在第一行可使每一次螢幕都顯示標題。第二行則只在程式開始執行時，顯示標題。

主流程

130	設定變數。
150	顯示標題及各欄標題。
180	計算目前R值的第一個S值。
190－210	計算A，B及C。
220－230	顯示一行數值，加計數器。
240	計算下一個S值；若無，則計算下個R值。
250－300	判斷是否R與S互質。
350－390	副程式，顯示標題及各欄的起頭。
400－460	檢查螢幕是否已滿。若如此，等待操作者按鍵。

主要變數

R，S	見"使用方法"的解釋
K	顯示的直角三角形總計數器
D	螢幕一畫面的顯行數計數器
A,B,C	直角三角形的三邊長

S1，B1，判斷R與S是否互質時所用的變數
R1，N

研究建議

1. 螢幕除了每一行顯示K，A，B與C外，再額外顯示R與S。也許您必須將各欄擠密一點。
2. 因爲程式所使用的整數值一直增大之後，慢慢超出 Apple 電腦九位準確度之能力，而使結果不再正確。您可否判斷出此狀況？而修改程式，使其在發生此現象時停止執行。

附錄 1　記憶體的使用

本書所有的程式都能在 Apple Ⅱ 標準型、Apple Ⅱ 擴充型、及 Apple Ⅲ 系統上執行。以下的圖表說明依系統具有的 RAM 記憶體、及 Applesoft BASIC 的種類分類出書中的程式可適合何種系統。注意，Apple Ⅱ 擴充型在 ROM 內有 Applesoft。

16K RAM	*32K RAM*	*48K RAM*	*Applesoft* 類型
Note 1	Note 2	Note 2	磁帶 Applesoft
所有	所有	所有	ROM Applesoft
Note 3	Note 4	All	ROM Applesoft 以及 Disk II (DOS 3.2.1)
(不可能)	Note 5	Note 2	DOS Applesoft 以及 Disk II (DOS 3.2.1)

NOTE 1：所有程式都可執行，除了 DECIDE、JOT、WALLOONS 與 GRAPH。

NOTE 2：所有程式都可執行，除了 WALLOONS。

NOTE 3：所有都可執行，但 DECIDE、GROAN

、JOT、WALLOONS、GRAPH與 STATS 執行時不可使用 DOS 軟體。

NOTE 4：所有都可執行，但WALLOONS執行時不可使用 DOS 軟體。

NOTE 5：除了WALLOONS以外都可執行，但 GRAPH 執行時不可使用 DOS 軟體。

有一些程式在執行之時必須依課文所述做一些修正。而且有些程式會受您的系統可用記憶空間的一些限制，此時就依Applesoft BASIC 所控制的可用空間而定，而且不要輸入別的 BASIC 程式。欲看您的系統目前有多少可用空間，可打入命令：

```
PRINT FRE(0)
```

Applesoft 取得控制後，例如，Cassette Applesoft 的 16K系統指出其有 4093 byte 可用空間。若顯示一個負值，其意爲可用空間大於 32,767 byte 。

書中的程式曾用DOS 3.3 以及DOS 3.2.1系統測試過。在出書之時，沒能在Apple Ⅲ 測試（在 Apple Ⅱ 模擬狀態下）。我們相信在此狀況下應能夠正確執行。

參考書目

書　籍

Bell, R. C., *Board and Table Games From Many Civilizations*, Oxford University Press, London, 1969. (WARI)

Brown, Jerald R., *Instant BASIC*, Dymax, Menlo Park, Calif., 1977. (Self-teaching text on the BASIC language)

Cohen, Daniel, *Biorhythms in Your Life*, Fawcett Publications, Greenwich, Connecticut, 1976. (BIORHYTHM)

Crow, E. L., David, F. A., and Maxfield, M. W., *Statistics Manual*, Dover Publications, New York, 1960. (STATS)

Croxton, F. E., Crowden, D. J., and Klein, S., *Applied General Statistics* (Third Edition), Prentice-Hall, Englewood Cliffs, N.J., 1967. (STATS)

Gruenberger, Fred J., and Jaffray, George, *Problems for Computer Solution*, John Wiley and Sons, New York, 1965. (BIRTHDAY, PI)

Gruenberger, Fred J., and McCracken, Daniel D., *Introduction to Electronic Computers*, John Wiley and Sons, New York, 1961. (MILEAGE, PI, PYTHAG, WARI as Oware)

Hildebrand, F. B., *Introduction to Numerical Analysis*, McGraw-Hill, New York, 1956. (CURVE, DIFFEQN, INTEGRATE, SIMEQN)

Kuo, S. S., *Computer Applications of Numerical Methods*, Addison-Wesley, Reading, Massachusetts, 1972. (CURVE, DIFFEQN, INTEGRATE, SIMEQN)

McCracken, Daniel D., and Dorn, W. S., *Numerical Methods and FORTRAN Programming*, John Wiley and Sons, New York, 1964. (CURVE, DIFFEQN, INTEGRATE, SIMEQN)

Shefter, Harry, *Faster Reading Self-Taught*, Washington Square Press, New York, 1960. (TACHIST)

期 刊

Feldman, Phil, and Rugg, Tom, "Pass the Buck," *Kilobaud*, July 1977, pp. 90-96. (DECIDE)

Fliegel, H. F., and Van Flandern, T. C., "A Machine Algorithm for Processing Calendar Dates," *Communications of the ACM*, October, 1968, p. 657. (BIORHYTHM)

H. K. $ 35.00

APPLE II BASIC 應用程式

葉耀明譯

協群出版社